Virtual Hands in VR: Motion Capture, Synthesis, and Perception

SIGGRAPH 2020 Course

Level: Intermediate

Moderator: Sophie J•rg, *Clemson University*
Lecturer: Yuting Ye, *Facebook Reality Labs*
Michael Neff, *University of California, Davis*
Franziska Mueller, *Max Planck Institute for Informatics*
Victor Zordan, *Clemson University*

Synopsis

This course presents the current state of the art about virtual hands with a focus on how to capture and synthesize hand motions for virtual reality. We furthermore show how hands are represented in current applications and summarize insights from perceptual studies on virtual hands.

Abstract

We use our hands every day: to grasp a cup of coffee, write text on a keyboard, or signal that we are about to say something important. We use our hands to interact with our environment and to help us communicate with each other without thinking about it. Wouldn't it be great to be able to do the same in virtual reality? However, accurate hand motions are not trivial to capture. In this course, we present the current state of the art when it comes to virtual hands. Starting with current examples for controlling and depicting hands in virtual reality (VR), we dive into the latest methods and technologies to capture hand motions. As hands can currently not be captured in every situation and as constraints stopping us from intersecting with objects are typically not available in VR, we present research on how to synthesize hand motions and simulate grasping

SIGGRAPH '20 Courses, August 17, 2020, Virtual Event, USA
ACM 978-1-4503-7972-4/20/08.
10.1145/3388769.3407494

motions. Finally, we provide an overview of our knowledge of how virtual hands are being perceived, resulting in practical tips on how to represent and handle virtual hands.

Our goals are (a) to present a broad state of the art of the current usage of hands in VR, (b) to provide more in-depth knowledge about the functioning of current hand motion tracking and hand motion synthesis methods, (c) to give insights on our perception of hand motions in VR and how to use those insights when developing new applications, and finally (d) to identify gaps in knowledge that might be investigated next. While the focus of this course is on VR, many parts also apply to augmented reality, mixed reality, and character animation in general, and some content originates from these areas.

Course Overview

15 minutes: Introduction

Welcome and Overview (5 Minutes)

Sophie Jörg
Overview of the course goals and motivations for attending. Speaker introductions.

Introduction to Virtual Hands (10 minutes)

Sophie Jörg
How are virtual hands currently used in applications? What are the standard devices for interactions (controllers), how are they represented? Examples of current games and applications. We furthermore introduce the basic anatomy of hands.

75 Minutes: Motion Capturing Fingers

Optical Marker-Based Approaches (25 Minutes)

Yuting Ye
Using markers to capture finger motions are proven to be challenging due to the small real estate on a hand comparing to the body. We will discuss commercial solutions in hand motion capture and how researchers tackle technical challenges such as automatic marker labeling and marker occlusion.

Gloves and Non-Optical Approaches (10 Minutes)

Michael Neff
Advantages and drawbacks of capturing hand motions with wearable sensors or gloves. Presenting different technologies and models.

Image- and Depth-Sensor-Based Approaches (40 Minutes)

Franziska Mueller
Recent advancements in deep learning enables marker-less motion capture of hands from images. These techniques greatly reduce the friction of capturing and using hand motions.

5 Minutes: Break

45 Minutes: Hand Motion Synthesis

Kinematic Hand Motion Synthesis (15 Minutes)

Michael Neff
How can we synthesize hand motions if we can not capture them? Introducing data-driven and procedural approaches.

Physical Modeling (30 minutes)

Victor Zordan
Interacting with objects in VR is still a challenge. Presenting current state and advances in hand object interaction and grasping.

25 minutes: Perception of Virtual Hands

Sophie Jörg
What are the consequences of having inaccurate hand motions? Discussing the effect of hand motions on the perception of virtual characters, the virtual hand illusion, the influence of different hand representations, and visualizations for virtual grasping, giving practical tips.

15 minutes: Conclusions and Q&A

Everyone

About the Lecturers

Sophie Jörg

School of Computing
Clemson University
sjoerg@clemson.edu
https://people.cs.clemson.edu/~sjoerg/

Sophie Jörg is an Associate Professor at Clemson University's School of Computing. Her research interests include character animation, motion perception, and virtual reality. She is fascinated by synthesizing motions for virtual characters and understanding their effects on the viewer. She has investigated different aspects of hand and finger motions such as capturing, analyzing, understanding, and automatically generating hand movements as well as using them for interaction. She holds a PhD from Trinity College Dublin and was an intern at Disney Research, Pittsburgh, and a postdoctoral researcher at Carnegie Mellon's Graphics Lab.

Yuting Ye

Facebook Reality Labs
yuting.ye@fb.com
http://yutingye.info/

Yuting Ye is a research scientist at Facebook Reality Labs. She works on human tracking and user interaction techniques to shape artificial (virtual, augmented, or mixed) realities as the future computing platform. Her recent research focuses on accurate skeletal hand tracking for virtual reality applications. She is especially interested in combining prior knowledge of human motion in deep learning problems. She holds a PhD in computer science from Georgia Institute of Technology on the simulation and control of virtual characters.

Michael Neff

Department of Computer Science and
Department of Cinema and Digital Media
University of California, Davis
mpneff@ucdavis.edu
https://web.cs.ucdavis.edu/~neff/

Michael Neff is a Professor in Computer Science and Cinema & Digital Media at the University of California, Davis where he leads the Motion Lab, an interdisciplinary research effort in character animation and embodied interaction. He holds a PhD from the University of Toronto and is also a Certified Laban Movement Analyst. His research interests include expressive character animation, gesture synthesis and understanding the impact of nonverbal communication on human interaction. He has studied how hand motion impacts personality perception, explored better techniques for glove-based capture and synthesized hand motion with a range of techniques for gesture and sign language.

Franziska Mueller

Max Planck Institute for Informatics
Saarland Informatics Campus
frmueller@mpi-inf.mpg.de
https://people.mpi-inf.mpg.de/~frmueller/

Franziska Mueller is a 4th-year PhD candidate in the Graphics, Vision and Video group at the Max Planck Institute for Informatics. During her PhD she has worked on several methods for real-time hand reconstruction from RGB and depth images, focusing on challenging scenarios like strongly occluded or interacting hands. She is especially interested in the combination of model-based techniques and machine-learning components. She holds a MSc in Computer Science from Saarland University.

Victor Zordan

School of Computing
Clemson University
vbz@clemson.edu
https://people.cs.clemson.edu/~vbz/

Victor Zordan is a Computer Science Professor and Administrator in Clemson University's School of Computing where he leads their Digital Arts program as well as their Visual Computing division. Victor's research focuses on the synthesis and analysis of movement for human characters and avatars as well as interaction metaphors, especially those pertaining to physical representations. He has focused on hand research in both synthesis through physical models as well as hand capture and automatically combining hand and full-body motion. Victor holds a PhD in computer science from Georgia Institute of Technology and gained tenure at the University of California, Riverside.

1 Introduction

1.1 Welcome and Overview

Course Motivation We use our hands every day: to grasp a cup of coffee, write text on a keyboard, or signal that we are about to say something important. We use our hands to interact with our environment and to help us communicate with each other without thinking about it. Wouldn't it be great to be able to do the same in virtual reality? However, hand motions are detailed and hands have many degrees of freedom, which is why their accurate motions are not trivial to capture. In this course, we present the current state of the art when it comes to virtual hands. Starting with current examples for controlling and representing hands in virtual reality (VR), we dive into the latest methods and technologies to motion capture and represent hands in VR. As hands can currently not be captured in every situation and as constraints stopping us from intersecting with objects are typically not available in VR, we present research on how to synthesize hand motions and simulate grasping motions. Finally, we provide an overview of our knowledge of how virtual hands are being perceived, resulting in practical tips on how to represent and handle virtual hands.

As technology evolves quickly in this field and many people join it and need to get to the current state of the art, this course will provide a great foundation and appeal to a broad public.

Intended Audience Our target audience includes researchers at all levels, developers especially of motion capture devices, designers of virtual reality (VR) applications, as well as users of games and applications in virtual environments. Several topics are accessible for a wider audience without previous knowledge, such as current methods of interactions with virtual hands, the perception of virtual hands including the virtual hand illusion, as well as some of the basics of motion capturing hands. Other parts, are aimed for an audience with some general knowledge in computer graphics, such as newer motion capturing techniques using deep learning or motion synthesis methods using physics-based animation.

Prerequisites While some parts of our course are accessible to a broad audience, other parts are aimed for an audience with some general knowledge in computer animation. For example, newer motion capturing techniques use deep learning and some motion synthesis methods use physics-based animation or motion graphs.

1.2 Introduction to Virtual Hands

Using your own hands in virtual reality: what once was pure science fiction is becoming a reality for consumers. While the hardware to experience virtual reality (VR) and to capture the motions of one's own hands used to be very expensive, the equipment is now more affordable and has been spreading beyond research labs and industry facilities for a few years. New hardware is entering the market every year and more and more games and applications are offered. However, while hand tracking is possible, most commercial applications use controllers to interact with the virtual environments.

Despite a lot of recent progress, accurately motion capturing hands remains a challenge [130]. Doing so in real-time without expensive equipment is even more challenging.

Hands have a high number of degrees of freedom and are smaller relative to the body. They have a complex anatomical structure and parts of a hand can be occluded. All these reasons make it a difficult task.

An additional issue that arises when tracking and visualizing one's real hands in VR, aside from accurate tracking, is the timely recognition of grasp actions. A controller has the advantage of having buttons that allow for reliable detection of any grasp from the user. When interacting with tracked hands, however, a grasp has to be inferred from the user's motions which might { depending on the algorithm used and the performance of the user { be slower or not always successful. Current research indicates that controllers can be more efficient when it comes to accomplishing a task that involves grasping compared to tracking hand motions directly using a glove with optical markers [55]. Still, in the presented study, gloves were preferred by the majority of participants for various reasons including that they were simply \more fun".

Finally, a general issue when using virtual hands, whether it is with a controller or with tracked hands, is how to represent interactions. If a user grasps an virtual object, there is no solid surface that keeps the user from intersecting with the geometry. A common solution or workaround to this problem is to hide the virtual hand geometry as soon as an object is being grasped, so that the player only sees a floating object. The game *Job Simulator* [76] is an example of such an implementation. A different solution is to automatically adjust the hand poses to match the geometry such as in the games *The Climb* [15] or *Lone Echo* [86].

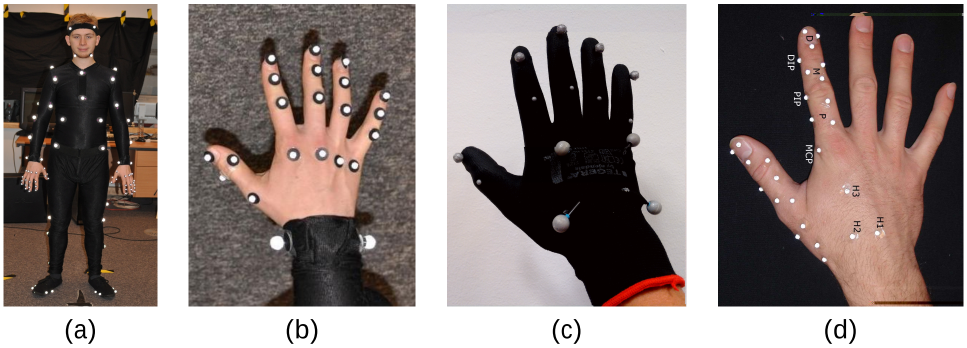
2 Motion Capturing Fingers

To use our own hands in virtual environments, we need to be able to track their motions in real time. In this largest part of the course, we describe different methods to capture the detailed motions of fingers.

2.1 Optical Marker-Based Approaches

Optical marker-based motion capture (mocap) of full body motions has been widely used in video games and special effects. Hand motions, especially fine-grained finger movements, are however more challenging to capture and therefore less utilized. The difficulty comes from the smaller size of hands comparing to the body, and their highly articulated nature. Large markers that work well on the body will need to cluster densely on a hand to capture all its articulations. A dense set of large markers are difficult to identify for tracking purpose. Large markers are also bulky and uncomfortable to put on, limiting the freedom of motion for the hand. As a result, smaller markers are usually used to capture more detailed and subtle hand motions (see Figure 1(a)). However, smaller markers are harder to identify robustly and easier to get occluded. Depending on the application, setting up motion capture for hands needs to consider marker sizes and density, camera resolution, camera coverage and their distance to the hands, and the desired motion detail. Once the 3D markers are captured, they need to be labeled according to the desired layout and calibrated against a suitable hand model. Finally, with labeled and calibrated 3D markers, an inverse kinematics (IK) problem can be solved to reconstruct the hand pose.

The suitable marker layout for finger capture depends on the specific applications. Dense marker layouts are more commonly used in biomedical research to analyze subtle



(a) (b) (c) (d)

Figure 1: Example marker sizes and layouts on the hand. (a) Smaller makers are used on the hands to capture subtle motions. They are harder to identify compared to larger markers used on the body. (b) A full marker set that can capture the movements of all finger joints. (c) A reduced marker set layout with mixed marker sizes [1] (©Alexanderson et al.). (d) A dense marker layout to capture subtle finger movements, usually to study the underlying biomechanical structure [13] (©Elsevier).

finger movements, especially for intricate thumb rotations [122, 121, 13]. In this case, multiple tiny markers are placed onto a single finger segment to capture its detailed rigid motion (see Figure 1(d)). For better marker visibility, the capture volume is usually limited and therefore the range of hand motion is constrained. On the other hand, a reduced marker layout (see Figure 1(c)) is often used together with body motion capture to animate virtual avatars. A sparse marker layout permits the use of larger marker sizes and less unconstrained movements in a larger capture volume.

While a reduced marker layout is easier for tracking markers in 3D, it is not sufficient to constrain all the degrees of freedom (DOFs) of the fingers. How to reconstruct plausible finger motions with incomplete observation becomes an interesting research question [37]. In addition, it is important to place the reduced marker set strategically so as to optimize the quality of the reconstructed motion. Schröder et al. [96, 95] devised an optimization procedure to compute marker placements such that they can best constrain the hand pose while minimizing occlusion. Kang et al. [46] utilized a dataset of the target motion to pick marker combinations that minimize the motion reconstruction error. Wheatland et al. [129] applied Principle Component Analysis (PCA) to rank and pick important markers based on the target motion content. Various results suggest six to eight markers are usually sufficient to capture conversational gestures or common poses in the American Sign Language (ASL).

Even with a reduced marker set, labeling each marker correctly remains a challenging task that usually requires tedious manual effort. Alexanderson et al. [1, 2] proposed an automatic marker labeling algorithm using multi-hypothesis tracking. They use Gaussian Mixture Models (GMMs) to represent the spatial distribution of marker positions local to the hand. The temporal transition probabilities of markers are modeled using Kalman filter [45]. With these two probability distributions, the most probable label assignment can then be selected using the Viterbi algorithm [25]. Recently, with the rapid advancement of deep learning techniques, even a full marker set can now be labeled automatically in real time using convolutional neural networks. Han et al. [31] formulated the marker labeling problem as an image regression problem. An unordered set of markers are rendered into a depth image. A convolutional neural network then predicts the 3D marker positions in the desired order from the input image. By matching the unordered input positions with the predicted ordered positions using bipartite matching, the marker labels are recovered.

Alternatively, the marker labeling problem can be solved by using active markers. These are LEDs that emit light at unique frequencies or phases to automatically identify themselves. In the past, only a small number of active markers are supported at the same time due to the frequency or phase limits, and the added electronics increase the friction of usage. However in recent years, active LED markers become more mature with smaller form factor, improved hardware specs, and smooth software integration from commercial products [75, 120, 79], making them an attractive finger tracking solution [5].

An inherent problem with optical motion capture is marker occlusion. It can be especially severe for heavy interactions, such as two-hand interaction or dexterous object manipulation. Pavllo et al. [77, 78] used a deep neural network to predict occluded active marker positions. An auto-encoder is trained to reconstruct marker positions by randomly omitting input markers, a procedure similar to dropout. The network therefore learns to fill in the missing markers through their correlations with the visible markers. While effective for sporadic occlusions, it cannot yet handle long occlusion period or when more markers are missing. A promising future direction is to explore sensor fusion between active makers and non-optical sensors. The next subsection will introduce alternative sensing techniques for capturing finger movements.

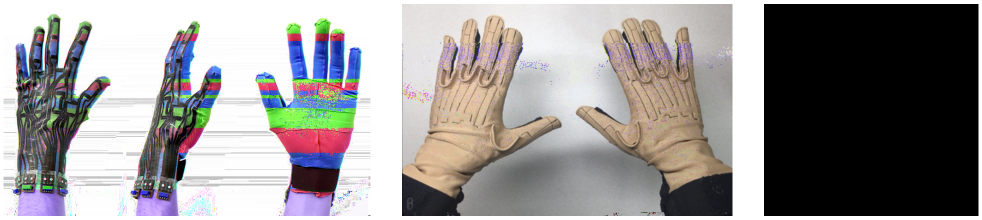
In conclusion, optical motion capture can accurately track subtle finger movements in real time, providing a high degree of embodiment and immersion for VR. However, due to the expensive equipment and dedicated space requirement, it is not a suitable consumer solution, but rather a time machine to explore the potential of accurate finger tracking in VR applications.

2.2 Gloves and Non-Optical Approaches

Non-optical hand tracking remains a significant focus in both research and commercial development, especially with the development of new materials with embedded sensing. Non-optical approaches normally use gloves, or occasionally straps or glue, to attach some form of sensor to the users' hands. The output of these sensors is then mapped to the joint angles of a hand skeleton. In contrast to optical marker-based motion capture approaches, these approaches offer the advantage of robustly providing data in any environment. They are not impacted by occlusions and there is generally no limitation on the capture volume. A disadvantage is that additional hardware is required. However, this can be augmented with additional devices that provide sensory feedback such as pseudohaptics.

Glove-based methods to capture hand motions became popular in the late 1980s, at first as an interface for gesture input for virtual environments [105]. Since then many different types and techniques have been proposed, some of them being commercially available, others remaining experimental prototypes (see Figure 2). Variation includes the type of sensor, number of sensors, accuracy, frame rate and calibration process. Gloves may also be wired or wireless and vary in size and weight of both the glove and required battery packs and transmitters.

Sensor configurations vary between a low of five sensors, one for each finger, to 22 sensors, one for each phalangeal joint with further sensors placed to record abduction



(a) (b) (c)

Figure 2: Examples of gloves: (a) Stretch-sensing glove with capacitive sensors presented at SIGGRAPH 2019 [29]; (b) Pair of CyberGloves 2 with 18 bend sensors [17]; (c) The glove Dexmo provides force feedback

and adduction of each digit as well wrist motion and the arch of the palm. Some recent work even employs more sensors than there are hand DOFs [29].

Below we will summarize the main sensor technologies used in gloves. More detailed surveys can be found in [85] and [20].

Bend Sensors: A traditional approach to glove design employs piezoresistive sensors that change their resistance as they are bent. This creates measurable voltage changes that can be mapped to changes in joint angles. A design goal is to build linear sensors, which makes it easier to create the mapping between the signal and desired joint angle. The sensors are sewn into gloves that effectively position them over the joints of interest. The CyberGlove is a commercial system based on bend sensors [17].

Bend sensors can be effectively attached over flexion joints, such as the proximal interphalangeal joints. This provides a one-to-one mapping between the joint movement and sensor reading that allows for accurate measurement. It is more difficult to accurately measure joints involved in abduction and adduction. Joint movement will generally impact multiple sensors and this cross-talk can lead to lower quality reconstruction and/or the need for more complex calibration processes.

Stretch Sensors: Recent approaches have explored using materials that provide a changing signal as they stretch. These materials may change in either resistance or capacitance. Capacitive sensors are relatively small, allowing a large number to be placed on a glove. In recent work, this has exceeded the degrees of freedom of the hand [29]. A commercially available glove made of stretch sensors has been released by StretchSense [100].

IMUs: Inertial Measurement Units, or IMUs, consist of 3-axis accelerometers and gyroscopes, sometimes with the inclusion of magnetometers (e.g. [24]). These measure acceleration, rotational speed and orientation respectively. This data is integrated to track the IMU from a known starting configuration. IMUs are relatively low cost and can have very high sampling rates. The major drawback of IMUs is that because the signal is integrated, even small errors can accumulate, leading to drift. This can cause increasing error over time. Limiting this error has been a significant focus of research and development. The Perception Neuron glove is one commercial example of an IMU-based data glove [71].

Electromagnetic Tracking: This form of tracking consists of a transmitter that generates a low-frequency magnetic field and sensors consisting of three orthogonal coils used to measure relative magnetic flux. Each sensor provides 6-DOF position and orientation information, relative to the transmitter. The sensors have been miniaturized, making them suitable for hand tracking. Compared to marker-based motion capture, electromagnetic sensors do not need to be labeled and do not suffer from occlusions. The main issue with electromagnetic tracking is that magnetic interference is ubiquitous, which can warp the magnetic field and impact accuracy and/or calibration requirements. Polhemus is a commercial provider of magnetic tracking solutions [80].

Hybrid approaches: Several approaches have tried to combine the strengths of different sensing technologies. For example, Rokoko combines IMUs with magnetic tracking [90]. ManusVR combines bend sensors with IMUs [61]. Combining approaches aim to mitigate the drift of IMUs, the poor abduction tracking of bend sensors and the interference of magnetic systems.

No matter the type of sensor that is used, a mapping must be constructed that goes from the sensor output to joint angles of the fingers, which are the desired output for most applications. Some devices will allow direct measurement of all finger DOFs. For others, inverse kinematics may be used, for example to calculate internal joint angles from the position of finger tips relative to the palm. Relationships between the DIP and PIP angles can also be enforced to reduce the active number of degrees of freedom [88].

A calibration process tunes the mapping from sensor signals to joint angles. Calibration must normally be done on a per-subject basis as hand size and finger length vary. Sometimes this is done every time a glove is worn to accommodate any variations in fit. It is desirable, therefore, for the calibration process to be fast. Calibration may prioritize hand shape or accuracy in end effector position, to for example ensure that if a finger and thumb touch in real life, they touch in the reconstruction. Discrepancies between hand shape and end effector position can arise even with accurate angle measurements if the proportions of the finger in the skeleton model are not an exact match for those of the subject.

Traditional calibration approaches rely on sensors being approximately linear so that calibration can be done with a small number of poses; as few as two. This allows for fast and easy calibration. Newer techniques have grown more algorithmically complex and require more data (e.g. [127]). This increase in complexity and calibration time provides more accurate reconstructions. Recently, neural networks have been employed with a million samples (e.g. [29]) to learn a more complex mapping function. The trained network can be customized to particular users.

If users are willing to wear gloves for a given application, this opens up the possibility of also incorporating haptic feedback into these devices. Some approaches add vibrotactile feedback to existing tracking technologies, such as bend sensors (e.g. CyberTouch [17]), or active LED markers (e.g. ART [5]). Other approaches use mechanical exoskeletons to provide force feedback (e.g. Dexamo [89] and Haptx [32]).

Glove systems share many of the same advantages and disadvantages. A leading advantage is that they are essentially impervious to occlusions. There are also no requirements for additional equipment to be mounted in the capture space and the working volume is very flexible. As discussed, there are a wide range of options.

Disadvantages include that users must wear a device, providing extra encumbrance and set up time. Calibration may also take some time. Accuracy varies and it is difficult to match the accuracy of optical motion capture. Most gloves do not provide global position, and orientation relying on another tracking solution for this. While cheaper than optical motion capture systems, gloves still generally run in the thousands of dollars, making them inaccessible for many home users.

2.3 Image- and Depth-Sensor-Based Approaches

Image- and depth-sensor-based approaches for hand motion capture have been actively researched since they alleviate the need for instrumentation of the hand | in contrast to marker-based or glove-based approaches.

Earlier works have employed multi-camera setups consisting of multiple calibrated RGB and/or depth cameras to leverage multi-view constraints [6]. However, calibration of these setups is tedious and may be hard for non-expert users. Furthermore, calibrated setups are inflexible and hence unsuitable for applications in VR/AR or mobile devices. Therefore, more recent approaches have focused on hand motion capture from a single RGB [66, 102, 139, 9] or depth camera [133, 106]. Products have been developed in this area such as the Leap motion [51] or the Oculus Quest [73].

Approaches for image-based hand tracking can in general be divided into three categories:

1. generative or model-based,
2. discriminative or data-driven,
3. hybrid.

2.3.1 Generative Methods

Generative methods make use of a hand model and optimization techniques to reconstruct hand motion from observations in images.

Modeling the Hand. Hand models typically have an underlying kinematic skeleton, a hierarchy of transforms which corresponds to the joints and bones of the hand. The output of model-based hand motion capture methods are model parameters which describe the configuration of the hand model, namely rigid transform, hand pose, and occasionally hand shape. A hand model further needs to model the surface or the volume of the hand since these are observable in the input images, in contrast to just the bones. Many different hand models have been successfully used in the past, e.g., collections of geometric primitives [74], surface meshes [118, 92], sphere meshes [115], subdivision surfaces [112], articulated distance functions [113], or Sum of Gaussians models [104].

Optimization-based Model Fitting. Generative methods are based on the analysis-by-synthesis principle. The hand model at the current parameter hypothesis is compared to the observation in the input image. The comparison uses an objective function which is designed to measure discrepancy between the model and the input image. Optimization techniques are used to minimize the objective function and hence find the set of parameters which produces the best fit to the input observation. The objective function usually consists of several terms which can be divided into data terms and regularizers. Data terms compare the model at the current hypothesis to the input, e.g., based on silhouettes, point cloud alignment, or color similarity. Regularizers are priors that encourage plausible results and include, for example, temporal smoothness [66, 107], joint angle limits [115, 103], pose subspaces [92, 67, 109, 107], and interpenetration avoidance [118, 67]. Especially when only a single RGB or depth camera is used, prior knowledge can help to resolve ambiguities. Various optimization techniques have been used for hand motion capture, e.g., gradient descent, Gauss-Newton, Particle Swarm Optimization. Depending on the optimizer, the generative method is more or less dependent on a good initial parameter hypothesis.

2.3.2 Discriminative Methods

Discriminative methods, in contrast to generative methods, often run independently per frame and do hence not depend on good initialization. They make use of a large data corpus to either build a database for pose look-up [91] or train machine learning components, like random decision forests [47, 123, 110] and — in more recent years — neural networks [116]. Especially when using depth images, many different choices for input representations have been explored, including images, point clouds [27], and voxel grids [64]. The prediction tasks differ across the literature. Some methods perform a per-pixel classification of the image into hand parts and obtain joint positions in a post-processing step [47]. Others estimate 3D joint positions directly and in addition minimize the projection error in the image to encourage accurate results in both 2D and 3D [66]. In this course, we discuss variations of deep learning techniques for hand pose estimation.

Input Encoders. Given an input RGB or depth image, the first part of the neural network, the so called *input encoder*, processes the input image to obtain an abstract feature representation. There are different possible choices of input encoders, some depending on the input modality (RGB or depth). While there are many popular architectures for neural networks [33, 72], we focus on core operations and building blocks here.

The widely used *2D convolutional neural networks* (2D CNNS) employ convolution filters in image space to process an input image [21]. An input image with width w, height h, and c channels is encoded to a feature representation of size $\frac{w}{x} \times \frac{h}{y} \times f$, where f is a chosen number of feature maps. Since 2D CNNs operate on images, they can be applied to both RGB and depth images.

In contrast, *3D CNNs* require volumetric input data, as obtained for example from a depth image [63]. Starting from a voxel grid of resolution $w \times h \times d$ containing the input data, volumetric convolutions are applied to obtain a feature representation of size $\frac{w}{x} \times \frac{h}{y} \times \frac{d}{z} \times f$ [64]. While 3D CNNs directly encode data in 3D, they generally have a larger memory requirement than 2D CNNs.

A *PointNet* is another possible input encoder that is designed for 3D input data, specifically pointclouds [82]. Starting from a set of n points, each represented by their 3D coordinates and possibly other features (e.g. normal or color), a Multi-Layer Perceptron is used to obtain a feature representation of size $n \times f$ from which a global feature can be extracted by a pooling operation. To capture local structures better, the hierarchical PointNet has been proposed as an extension, where the features are aggregated in a local neighborhood before being processed by the next level PointNet [83, 27].

Output Representations. Once the input RGB or depth image has been encoded, the resulting feature representation is further processed to obtain the desired output, the 3D hand pose. Note that for monocular RGB data, there is always an inherent scale ambiguity, i.e., the 3D hand pose in world coordinates can only be calculated up to a single scale factor. This has to be properly handled in the output, e.g., by regressing normalized 3D coordinates relative to a reference joint. Although regression of absolute

3D coordinates is possible with depth input data, some kind of normalization is often used to reduce the variance in the input and output data, making learning easier. There are different choices for representing the output 3D hand pose.

The most straightforward output representation are *vectorized joint positions* [139, 66]. Starting from the feature representation, linear layers, possibly in combination with non-linearities, are applied to obtain the output of size $N_J \times 3$, where N_J is the number of joints. Due to the use of the linear layers, the output usually lies on a low-dimensional manifold of plausible 3D hand poses. Note that what the network learns as \plausible" depends on the training data and can easily lead to biases if the whole hand pose space is not equally covered. Furthermore, predicted vectorized joint positions might not project well onto the input image without the use of an explicit reprojection loss [10, 66].

Hence, several output representations that are closer to the input image have been proposed. For *2D joint heatmaps*, the network produces a single-channel output image for each joint [116]. This output heatmap shows where the network believes the joint to be in the input image. Ideally, the heatmap for each joint should only contain a single peak and be 0 everywhere else. A \smudged" heatmap, i.e., where the values are spread out, usually indicates that the network is not confident in its prediction for this joint. The spatial size of the heatmaps can be the same as the original input image or a fraction thereof. There is a trade-off between spatial accuracy and runtime/memory cost. To go from the condensed feature representation to the output heatmaps, transposed convolutions are used [21]. 2D heatmaps only encode 2D hand pose. To obtain the 3D hand pose, the 2D hand pose can be lifted by a separate network [139] or a kinematic skeleton can be fitted to the 2D hand pose [116]. The concept of heatmaps can be generalized to 3D input and output, where the input is a 3D voxel grid and the output is a *3D heatmap grid per joint* [64].

Furthermore, 2D heatmaps can be combined with other output representations in order to represent a 3D hand pose. *Location Maps* [62] consist of 3 additional output images per joint, encoding the 3D x, y, and z position or offset from a reference joint. Once the 2D location of a joint is determined (e.g., by non-maxima suppression in the heatmap), the respective location maps can be read out at that 2D location to obtain the 3D pose.

Another image-based extension to 2D heatmaps are *3D Part Orientation Fields* [58, 131], which are based on 2D Part Affinity Fields [12]. Per bone, 3 additional output images are predicted that form a vector field and describe the 3D orientation of the bone. Given the 2D locations of the two joints that form the bone as predicted in the heatmaps, the x,y, and z values for the 3D orientation can be read from the vector field.

2.3.3 Hybrid Methods

To combine the best of both worlds, the discriminative and generative methods, many works have investigated hybrid methods [111, 67]. Earlier hybrid methods were often in large parts still generative model-based optimization systems. They employed discriminative components to obtain better initializations for the optimizer [84, 113] or to enhance the objective function with predictions like hand parts or salient points [103, 118]. With the advent of deep learning, the capacity of discriminative methods has greatly improved. Therefore, some more recent methods rely mostly on the output of a neural network, like joint locations, and fit a kinematic skeleton to these predictions in an optimization-based framework [116, 68]. This additional fitting step ensures temporally consistent and physically plausible poses. During the last years, some approaches have taken one step further and directly integrated hand models into neural networks layers. These networks usually regress hand model parameters and the objective function terms established in generative model-fitting frameworks can readily be used as losses for end-to-end training [19, 125]. Neural networks trained in this way learn to explain the input observations and are hence more robust to possible annotation biases in training datasets.

3 Hand Motion Synthesis

So far in this course, we have covered a range of approaches to capture detailed finger motions, ranging from optical marker-based methods to various non-optical sensors and gloves to using images or depth sensors. However, sometimes none of these methods are adequate. The hands might be outside of the capture area, the motion may not match the constraints of the virtual environment or hand motion might be required for non-tracked characters. It is still possible to create hand motions in these cases. In this part of the course, we will present several approaches that have been suggested to synthesize hand motions if only the body or wrist motions are available. The first section will focus on kinematic techniques that do not make reference on the underlying forces required to generate the motion and the second on the use of physical simulation to produce hand motion.

3.1 Kinematic Hand Motion Synthesis

	Kinematic	**Data-Driven**	**Physics-Based**
Non-Procedural	Keyframing	Motion Playback	Ragdoll
Procedural	Inverse Kinematics	Statistical Pose Models, Motion Repurposing	Controllers, Constrained Optimization

Table 1: A categorization of animation techniques with key exemplars of each.

Capturing hand motions for virtual reality has two main applications. One application is as an input modality. Here it is important to recognize a user's gestures in order to employ them to provide different forms of control input. While the base tracking technologies discussed here could be used as the input to such approaches, the input application is outside of the scope of this course. A second application is the generation of finger motion for characters, which is the focus of the material presented here. Characters can be divided into two broad categories. Avatars are direct representations of real people in a VR world. \Non-player characters" are other animated people in the world which are not projections of tracked users. Avatars have the additional requirement that their displayed movements must match the movements of the user. For self-presence, it can lead to a disconnect if a person sees their avatar moving differently to them. There is more flexibility when the avatar is displayed for other users, but inconsistency with the actual movement raises potential ethical and trust concerns.

Finger motion plays a central role in nonverbal communication and in manipulation of objects in the environment. Communication can roughly be divided into *gesture* and *sign languages*, although even self manipulations like scratching convey important information about a person's personality (e.g. [70]). Finger motion accompanying gesture generally involves more open-space movement where the fingers do not touch, but hand shapes like a fist or purse involve contact. Both the shape created by the fingers and the motion of the fingers can convey content. Sign languages rely on hand poses that can be relatively more complex and involve close interaction of the fingers. For example, the letter \T" in American Sign Language fingerspelling involves inserting the thumb between the index and middle finger while making a fist. Object manipulation relies on contact between the hand and the object. This contact can involve any or all of the fingers, ranging from the finger tip to the entire length of the finger. It may also involve the palm. Grasp is an important class of these manipulations which has been extensively studied. Other applications include touching other people and playing musical instruments. For the latter, both the design of the instrument and the music being played can define a sequence of required touch points.

For the purpose of this discussion, we will divide animation techniques into *kinematic*, *data-driven* and *physics-based*. Kinematic techniques describe motion in terms of positions and velocities, without reference to the underlying forces required to generate the motion. Data-driven techniques require motion data as input. These are also normally kinematic, but is useful to distinguish between data-driven and non-data based techniques (our base \kinematic" category). Physics-based techniques explicitly model the forces used to generate the motion and will be described in the next section. Each of these techniques can be split into *procedural* and *non-procedural* approaches. Procedural techniques rely on significant algorithmic work to generate the motion whereas non-procedural techniques do not. An overview is contained in Table 1

Non-procedural, Kinematic: Keyframing is the main technique in this category. In this approach, an artist specifies finger poses at key moments in time and controls the shape of interpolation curves that control the motion between these poses. Keyframing is appealing because it provides complete control over the motion, but this comes at the cost of potentially substantial manual labor, particularly for long sequences. Skilled animators can create very high quality motion and this can be customized for particular characters. The level of control has made it appealing for communication applications (e.g. [69]). In VR applications that rely on controller input, it can be a good option as a different hand shape can be specified for each button. It also can be useful where complex object interactions are required, such as gripping a gun [14]. Keyframes can also be a useful building block in algorithmic work.

Procedural, Kinematic: Inverse kinematics (IK) is the main procedural kinematic technique that is used for posing fingers in VR. IK takes as input a set of position and orientation constraints and solves for joint angles in order to achieve these. The most common formulation for IK is for constraints to be placed on the end effector of a kinematic chain. With hands, the palm position is often assumed to be set and IK is used to solve for the angles in each finger kinematic chain with these chains rooted at the palm. Some applications may also require contact with the palm or more parts of the finger than the end effector. Often rather small adjustments of the fingers are required in order to ensure contact with objects and avoid interpenetration.

A range of techniques have been proposed for solving inverse kinematics problems. For simple configurations of hinge joints, basic trigonometry can be sufficient. Heuristics like the two-thirds rule can simplify the problem [88]. Iterative approaches calculate the pseudo-inverse of the Jacobian at each pose which relates changes in joint angle to changes in position. They then take multiple small steps towards the solution [28].

Optimization [136] approaches based on sampling data [93] have also been developed.

Believable object interaction requires understanding the affordances of the object being manipulated. There are particular places where items are grasped (these can define IK targets), they may also have allowable orientations and particular ranges of motion. For example, a mug is held by its handle with the opening oriented upwards. A drawer moves in and out along its tracks and is pulled by its handle. Early work introduced \smart-objects" which knew their affordances and allowed pre-designed animations to be triggered for interaction in VR [44]. In work with musical instruments, the touch points are defined by the instrument design and the music being played [23, 138]. In general with these approaches, there is a tension between motion fidelity and remaining true to the actual user's movements. Triggering an animation may provide the best motion fidelity, but will likely not match the participant's motion. Small adaptations on top of actual tracking are often desired as a way to maximize immersion.

Grasp has emerged as an important game mechanic. For instance, \The Climb" offers a VR mountain climbing experience in which players must grasp rock holds. In \Lone Echo," players grasp objects in order to pull themselves through a space environment. Developers for this game built a system that would search for the environment object closest to the characters palm and then use simple IK, treating the fingers as hinge joints, in order to rotate the fingers to make contact [14]. Interestingly, they avoided physics due to concerns about computational cost and the need for graceful failure when players did penetrate objects with their hands.

Grasp also entails physical requirements, in that the grasp must be appropriate to support the weight of the object. This has been exploited by techniques that use the physical requirement of supporting an object in order to determine appropriate contact points for grasping it [132, 65]. The motion of the object is taken as input and the grasp is not physically simulated, but the forces required to support the object are used as a source of information to determine the correct contact points for the grasp.

Non-procedural, Data-driven: Sometimes finger motion is recorded to accompany a particular body motion and can simply be played back at runtime. This can provide excellent quality motion, but is not flexible and may require significant labor if many clips are required.

Procedural, Data-Driven: Grasp synthesis techniques have also made use of data-driven approaches. For example, Li et al. [53] build a motion capture database of different people grasping various objects. This implicitly defines the required touch constraints for a grasp. At runtime when the user wishes to pick up a new object, the system runs a shape match against the database to obtain potential grasp matches. These are clustered and then pruned based on a grasp quality measure, the resultant pose being used to generate the required animation.

A similar problem arises when people wish their avatars to touch in VR, say to shake hands or give a high five. For the motion to look plausible, the hands must touch, but not interpenetrate. This requires understanding appropriate contact points for these motions. Lee et al. [52] built a database of touch locations for such motions. At runtime, users' intended touch type can be identified and the implied touch locations can then be used to refine the animation of the remote participant in an avatar interaction.

Animation hand skeletons will often contain 20 DOFs or more per hand, leading to a high dimensional pose problem. In practice, however, there are many correlations between these DOFs and the \real" dimensionality of hand movement is much lower. Data has often been used to create lower dimensional embeddings, using either linear techniques like PCA (e.g. [129]), or nonlinear methods. A related issue is determining what do do with fingers that are not involved in a motion when only some fingers are required to satisfy touch constraints. ElKoura and Singh [23] encountered this situation when certain fingers were required to hold strings on a guitar and some were not. They used a collection of hand data and nearest neighbor search to determine the pose of uninvolved fingers.

A common process when dealing with fingers is "hand-over" animation. This reflects the traditional difficulty in tracking finger movement. Body and hand movement are recorded separately and then merged to create a final animation. Majkowska et al. [60] offered one of the early algorithmic approaches to this work. Hand and body motion were captured separately, in different capture volumes, but four markers were present in both captures. These markers were then used to time warp the hand motion to align better with the body motion.

While hand-over techniques were first envisioned for offline use, they may be adaptable to realtime applications in VR. This class of data-driven techniques follows a similar, three stage process. After motion is recorded, it is segmented into movement phases. These phases may correspond to phases of gesture behavior or grasps, for instance. At runtime, a match must be found between this pre-recorded data and the motion of the body. Nearest neighbor search, rule-based methods and optimization have all been used in this step. In a final phase, the recorded motion is adapted to best fit the body motion. This could be done with smoothing, blending, IK adjustment or physical simulation.

One application of these techniques has been to extend capture technologies. For example, colored gloves recorded by camera [126] or a reduced set of recorded markers [46, 129] were used to retrieve higher quality, full finger clips. In a communication application, Jörg et al. [42] followed the three stage approach to add finger data to body animations of gesture behavior by matching information about the arm and wrist movement. They experimentally validated the best matching heuristics.

3.2 Physical Modeling

Moving toward natural interface metaphors and generic manipulation with hands in VR, a promising path forward is the employment of physical simulation for hands, as in [81, 137, 48, 114, 18] among others. The key insight is that the force-based interaction afforded by physical simulation mimics the manner in which we affect the real world with our hands. However, there are a number of standing challenges, such as discrepancies between the real world and the simulated as well as limited controller sophistication, that must be addressed to make the technique useful in everyday VR.

In the recent advances in hand simulation, two distinct areas are important to hands in VR. One focus has been physical modeling and control of the articulated skeleton. This type of hand simulation holds promise of generic but nuanced hand interaction through the "virtual" physical interaction of the human user (driving a virtual hand) and the virtual environment. However, along with these benefits come a set of problems especially pertaining to assumptions of rigid-body hands and unrealistic contacts that dictate how virtual hands can be controlled and how they interact with objects in the virtual world. The second form of simulation aims to model an anatomically based simulation. Here the focus is on the physical and biomechanical components that create realistic appearances of the hand. Because of the complexity of anatomical hand simulation, we have not seen much overlap between the two topics to date, although the areas are complementary.

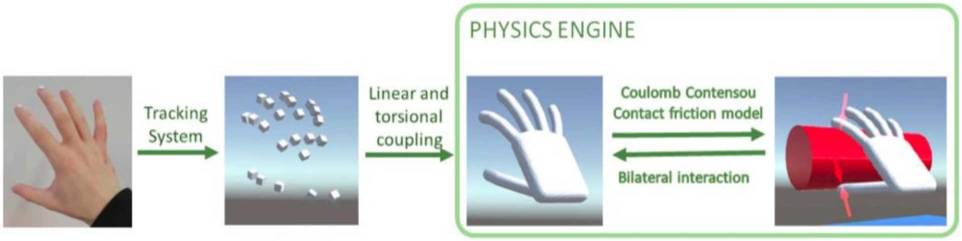

Tracking System
Linear and torsional coupling
PHYSICS ENGINE
Coulomb Contensou Contact friction model
Bilateral interaction

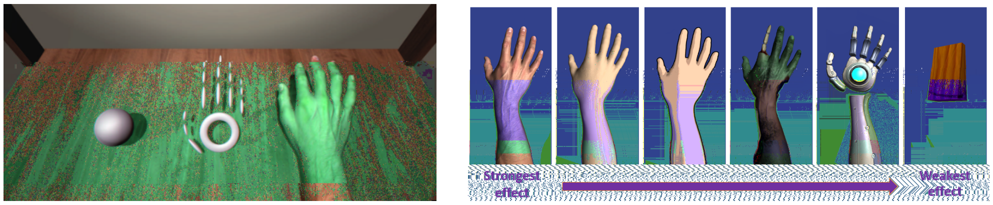
Figure 3: An example system for using physics in VR, image from Delrieu et al. [18] ©IEEE.

Often in the latter, a kinematic skeleton drives the physical model, while in the former, a simplified surface model is most often used to promote interactivity.

As real-world hands are the super instrument by which humans manipulate their world, building humanoid avatar, character and robotic hands has been a key focus in support of natural interaction. As in Figure 3, physical manipulation modeling aims to address motion planning coupled with control for joint articulation in order to build systems that are capable of the span of behaviors hands see everyday. As such, physical systems for hand grasping has been the focus of research for many years, as [81, 56, 65, 137]. Much of this work aims to solve the control for general manipulation through grasp shape and approach planning [53], coordinated compliant control [81], and dexterous manipulation [57, 3]. Motion data examples, libraries, and heuristics as well as optimization and automation techniques have been assembled in various capacities in the exploration of directable and believable simulated hand motion. For example, we have seen specialised sensor-based \interaction capture" performed to extract model parameters that are employed to create manipulation for physical interaction [50].

Virtual physical hands have been developed for a broad array of manipulation applications. In recent years, a focus growing in attention has become *interactive* physical models with an aim set on Virtual Reality applications [48, 35, 119]. Intuitive activities in VR enhance engagement and interactivity that contribute to enriched user experiences. However, the development of interfaces that bridge the gap between real-world input and virtual interpretation remains challenging due to a number of factors [39, 18, 114, 35]. Foremost, speed is key for real-time performance. However, there is also the inevitable discrepancy between objects that may appear in the virtual world, but that do not have presence in the physical. Further, while haptic devices hold appeal [38], their implementation and applicability remain limited. Thus, there is a feedback gap between the virtual and real that make it difficult to judge and correct actions as well as foil believability and responsiveness when considering a wide range of actions [11].

There has been some work in using simulated deformation, e.g. finite element modeling (FEM), to improve hand contact in VR [39, 108, 34]. A variety of FEM approaches have appeared in the literature [49, 26, 124]. The goal here is to use volume based deformation to generate the bulging, wrinkling, and stretching of the skin surrounding the hand as well as its underlying biological structure under different settings. As an example, recent work in this form of hand simulation supports deformation with FEM which has been fit to MRI data to create high fidelity hand shape and deformation using a kinematically animated skeleton [124]. The related work in support of VR primarily



(a) (b)

Figure 4: Examples of virtual hand models that have been used in experiments investigating the virtual hand illusion (a) from Argelaguet et al. 2016 [4] (©IEEE); (b) from Lin and Jörg [54]

highlights contact deformation, in an effort to synthesize better grasps, by specifically building deformation in the finger tips which increases contact area when the virtual hand touches an object.

4 Perception of Virtual Hands

After discussing approaches to capture hand motions and methods to synthesize them, a question that remains is: What happens if hand motions are not tracked accurately or not at all? Do we notice errors in hand motions? Do they affect the impression other people have from a character? Do our efforts to visualize hand motions accurately make a difference?

Previous research has shown that people are very good at perceiving subtle motions. When it comes to body motions, we know that we can recognize a walk within a tenth of a second even if we just see a few points attached to a person [40]. Based on so called point-light walkers, we also know that we can even identify a friend, just based on the way they move [16]. Experiments have indeed shown that errors in hand motions of a virtual character can be perceived and that they can even change the interpretation of a scene. For example, we can perceive that body and finger motions are desynchronized, even if they are just out of sync by 0.1 seconds (however, it depends on the motion) and delays in finger motions of 0.5 seconds can even change how we interpret a scene [41]. Furthermore, the detailed motions of the fingers alter the way we perceive the personality of a virtual character [128].

When it comes to perceiving our own hands in a virtual environment, the virtual hand illusion (VHI) becomes important. The VHI is a body ownership illusion in virtual reality that is similar to the rubber hand illusion (RHI). Botvinich and Cohen [8] showed in an experiment that participants, when they saw an object touching a simple rubber hand and synchronously felt the sensation of the touch on their own hand, report a sensation that the rubber hand is part of their body. This type of experiment - inducing a feeling of ownership for a rubber hand or similar objects through synchronized touch and visual feedback - has been repeated many times also with alterations when it comes to objects and procedures and is known as the rubber hand illusion [22, 117].

While some studies have reported that participants have described a certain degree of ownership over objects that are not hand-shaped or even over empty spaces [30], it seems that a resemblance of the object to a hand is typically required for the illusion to occur [87]. It has been shown that active or passive motion can also induce the illusion

[43]. Reported onset times for the illusion to occur vary between about 10 to 110 seconds and might depend on the details of the study as well as the selection of participants [87].

The VHI illusion is a similar effect in a virtual environment, where the rubber hand is replaced by a tracked virtual hand [101, 134]. The feeling that the virtual hand is part of one's body now comes from the synchrony between visual and proprioceptive information together with motor activity [94]. While there are still many open questions about the details of this illusion, experiments have shown that the virtual hand illusion can be generated for a variety of hand appearances as well as for more abstract objects. Examples include a square that changes in size or color [59], a balloon [59], a cat claw [135], abstract hands [4, 99], and hands with more or less fingers [36, 97]. However, the effect is stronger for more realistic hand models, see Figure 4 [4, 54]. It also seems that results vary widely between participants, with some participants feeling a strong sense of ownership even for abstract models while others do not feel any effect even for a realistic hand. One study found that female participants preferred female hands whereas male participants accepted avatar hands of both genders [98].

While we can not add all possible solid objects to the real world to avoid that our virtual hands intersect with virtual geometries, we can create user feedback for these cases. Especially when grasping, user feedback might be important to increase speed. Borst et al. [7] and Canales et al. [11] have investigated a series of visualizations for grasping interactions. Amongst other results, Canales et al. found that visualizing a tracked hand leads to the best performance among the tested options. Still, on average users preferred visualizations that prevent hand-object interpenetrations whereas hiding the virtual hand when grasping was liked least and reduced ownership of the virtual hand.

References

[1] Alexanderson, S., OSullivan, C., and Beskow, J. Real-time labeling of non-rigid motion capture marker sets. *Comput. Graph. 69*, C (Dec. 2017), 59–67.

[2] Alexanderson, S., O'Sullivan, C., and Beskow, J. Robust online motion capture labeling of finger markers. In *Proceedings of the 9th International Conference on Motion in Games* (2016), MIG '16, p. 7–13.

[3] Andrews, S., and Kry, P. G. Goal directed multi-finger manipulation: Control policies and analysis. *Computers & Graphics 37*, 7 (2013), 830–839.

[4] Argelaguet, F., Hoyet, L., Trico, M., and Lecuyer, A. The role of interaction in virtual embodiment: Effects of the virtual hand representation. In *2016 IEEE Virtual Reality (VR)* (March 2016), pp. 3–10.

[5] ART. Advanced Realtime Tracking website, https://ar-tracking.com/products/interaction/fingertracking/, [Online; accessed 1-July-2020].

[6] Ballan, L., Taneja, A., Gall, J., Gool, L. V., and Pollefeys, M. Motion Capture of Hands in Action using Discriminative Salient Points. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2012), Springer, pp. 640–653.

[7] Borst, C., and Indugula, A. Realistic virtual grasping. In *IEEE Virtual Reality Conference* (04 2005), pp. 91–98.

[8] Botvinick, M., and Cohen, J. Rubber hands `feel' touch that eyes see. *Bulletin of the Psychonomic Society 391*, 756 (1998).

[9] Boukhayma, A., Bem, R. d., and Torr, P. H. 3D Hand Shape and Pose From Images in the Wild. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), IEEE, pp. 10843–10852.

[10] Brau, E., and Jiang, H. 3D Human Pose Estimation via Deep Learning from 2D Annotations. In *Proceedings of the International Conference on 3D Vision (3DV)* (2016), IEEE, pp. 582–591.

[11] Canales, R., Normoyle, A., Sun, Y., Ye, Y., Luca, M. D., and Jörg, S. Virtual grasping feedback and virtual hand ownership. In *ACM Symposium on Applied Perception 2019* (2019), pp. 1–9.

[12] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), IEEE, pp. 7291–7299.

[13] Chang, L. Y., and Pollard, N. S. Robust estimation of dominant axis of rotation. *Journal of Biomechanics 40*, 12 (2007), 2707 – 2715.

[14] Copenhaver, J. Vr animation and locomotion systems in lone echo. In *Proceedings of GDC* (2017).

[15] CryTek, Oculus. The Climb website, https://www.theclimbgame.com/, [Online; accessed 1-July-2020].

[16] Cutting, J., and Kozlowski, L. Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin of the Psychonomic Society 9*, 5 (1977), 353{356.

[17] CyberGlove Systems. CyberGlove II website, http://www.cyberglovesystems.com/ cyberglove-ii, [Online; accessed 7-Feburary-2020].

[18] Delrieu, T., Weistroffer, V., and Gazeau, J. P. Precise and realistic grasping and manipulation in virtual reality without force feedback. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2020), pp. 266{274.

[19] Dibra, E., Wolf, T., Oztireli, C., and Gross, M. How to Refine 3D Hand Pose Estimation from Unlabelled Depth Data? In *International Conference on 3D Vision (3DV)* (2017), IEEE, pp. 135{144.

[20]

[29] Glauser, O., Wu, S., Panozzo, D., Hilliges, O., and Sorkine-Hornung, O. Interactive hand pose estimation using a stretch-sensing soft glove. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH) 38*, 4 (2019).

[30] Guterstam, A., Gentile, G., and Ehrsson, H. The invisible hand illusion: Multisensory integration leads to the embodiment of a discrete volume of empty space. *Journal of Cognitive Neuroscience 25* (04 2013).

[31] Han, S., Liu, B., Wang, R., Ye, Y., Twigg, C. D., and Kin, K. Online optical marker-based hand tracking with deep labels. *ACM Trans. Graph. 37*, 4 (July 2018).

[32] HaptX. website, https://haptx.com/, [Online; accessed 1-July-2020].

[33] He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), IEEE, pp. 770{778.

[34] Hirota, K., and Tagawa, K. Interaction with virtual object using deformable hand. In *2016 IEEE Virtual Reality (VR)* (2016), pp. 49{56.

[35] Höll, M., Oberweger, M., Arth, C., and Lepetit, V. Efficient physics-based implementation for realistic hand-object interaction in virtual reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2018), IEEE, pp. 175{182.

[36] Hoyet, L., Argelaguet, F., Nicole, C., and Lécuyer, A. \wow! i have six fingers!": Would you accept structural changes of your hand in vr? *Frontiers in Robotics and AI 3* (2016), 27.

[37] Hoyet, L., Ryall, K., McDonnell, R., and O'Sullivan, C. Sleight of hand: Perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2012), I3D '12, Association for Computing Machinery, p. 79{86.

[38] Humberston, B., and Pai, D. K. Hands on: interactive animation of precision manipulation and contact. In *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2015), pp. 63{72.

[39] Jacobs, J., and Froehlich, B. A soft hand model for physically-based manipulation of virtual objects. In *2011 IEEE Virtual Reality Conference* (2011), pp. 11{18.

[40] Johansson, G. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics 14*, 2 (1973), 201{211.

[41] Jörg, S., Hodgins, J., and O'Sullivan, C. The perception of finger motions. In *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization* (New York, NY, USA, 2010), APGV '10, ACM, pp. 129{133.

[42] Jörg, S., Hodgins, J. K., and Safonova, A. Data-driven finger motion synthesis for gesturing characters. *ACM Transactions on Graphics 31*, 6 (2012).

[43] Kalckert, A., and Ehrsson, H. H. Moving a rubber hand that feels like your own: A dissociation of ownership and agency. *Frontiers in Human Neuroscience 6* (2012), 14.

[44] Kallmann, M., and Thalmann, D. Direct 3d interaction with smart objects. In *Proceedings of the ACM symposium on Virtual reality software and technology* (1999), pp. 124{130.

[45] Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering 82*, 1 (03 1960), 35{45.

[46] Kang, C., Wheatland, N., Neff, M., and Zordan, V. Automatic hand-over animation for free-hand motions from low resolution input. In *Motion in Games* (2012), pp. 244{253.

[47] Keskin, C., Kiraç, F., Kara, Y. E., and Akarun, L. Hand Pose Estimation and Hand Shape Classification Using Multi-Layered Randomized Decision Forests. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2012), Springer, pp. 852{863.

[48] Kim, J.-S., and Park, J.-M. Physics-based hand interaction with virtual objects. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (2015), IEEE, pp. 3814{3819.

[49] Kry, P. G., James, D. L., and Pai, D. K. Eigenskin: real time large deformation character skinning in hardware. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2002), pp. 153{159.

[50] Kry, P. G., and Pai, D. K. Interaction capture and synthesis. *ACM Transactions on Graphics (TOG) 25*, 3 (2006), 872{880.

[51] LeapMotion. Leap Motion Controller, website, https://www.ultraleap.com/product/leap-motion-controller/, [Online; accessed 1-July-2020].

[52] Lee, Y., Lee, S., and Lee, S.-H. Multifinger interaction between remote users in avatar-mediated telepresence. *Computer Animation and Virtual Worlds 28*, 3-4 (2017), e1778.

[53] Li, Y., Fu, J. L., and Pollard, N. S. Data-driven grasp synthesis using shape matching and task-based pruning. *IEEE Transactions on visualization and computer graphics 13*, 4 (2007), 732{747.

[54] Lin, L., and Jörg, S. Need a hand? how appearance affects the virtual hand illusion. In *Proceedings of the ACM Symposium on Applied Perception* (New York, NY, USA, 2016), SAP '16, Association for Computing Machinery, pp. 69 | -76.

[55] Lin, L., Normoyle, A., Adkins, A., Sun, Y., Robb, A., Ye, Y., Di Luca, M., and Jörg, S. The effect of hand size and interaction modality on the virtual hand illusion. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2019), IEEE, pp. 510{518.

[56] Liu, C. K. Synthesis of interactive hand manipulation. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), pp. 163{171.

[57] Liu, C. K. Dextrous manipulation from a grasping pose. In *ACM SIGGRAPH 2009 papers*. ACM New York, NY, USA, 2009, pp. 1{6.

[58] Luo, C., Chu, X., and Yuille, A. OriNet: A Fully Convolutional Network for 3D Human Pose Estimation. In *Proceedings of the British Machine Vision Conference (BMVC)* (2018).

[59] Ma, K., and Hommel, B. Body-ownership for actively operated non-corporeal objects. *Consciousness and Cognition 36* (2015), 75{86.

[60] Majkowska, A., Zordan, V. B., and Faloutsos, P. Automatic splicing for hand and body animations. In *2006 ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (Sept. 2006), pp. 309{316.

[61] Manus. website, https://www.manus-vr.com/, [Online; accessed 1-July-2020].

[62] Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., Xu, W., Casas, D., and Theobalt, C. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. *ACM Transactions on Graphics (TOG) 36*, 4 (2017).

[63] Milletari, F., Navab, N., and Ahmadi, S.-A. V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In *Proceedings of the International Conference on 3D Vision (3DV)* (2016), IEEE, pp. 565{571.

[64] Moon, G., Yong Chang, J., and Mu Lee, K. V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), IEEE, pp. 5079{5088.

[65] Mordatch, I., Popović, Z., and Todorov, E. Contact-invariant optimization for hand manipulation. In *Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation* (2012), Eurographics Association, pp. 137{144.

[66] Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., and Theobalt, C. GANerated hands for real-time 3D hand tracking from monocular RGB. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2018), IEEE, pp. 49{59.

[67] Mueller, F., Davis, M., Bernard, F., Sotnychenko, O., Verschoor, M., Otaduy, M. A., Casas, D., and Theobalt, C. Real-time Pose and Shape Reconstruction of Two Interacting Hands with a Single Depth Camera. *ACM Transactions on Graphics (TOG) 38*, 4 (2019), 1{13.

[68] Mueller, F., Mehta, D., Sotnychenko, O., Sridhar, S., Casas, D., and Theobalt, C. Real-time Hand Tracking under Occlusion from an Egocentric RGB-D Sensor. In *Proceedings of the International Conference on Computer Vision (ICCV)* (2017), IEEE, pp. 1163{1172.

[69] Neff, M., Kipp, M., Albrecht, I., and Seidel, H.-P. Gesture modeling and animation based on a probabilistic re-creation of speaker style. *ACM Transactions on Graphics 27*, 1 (Mar. 2008), 5:1{5:24.

[70] Neff, M., Toothman, N., Bowmani, R., Fox Tree, J., and Walker, M. Don't scratch! self-adaptors reflect emotional stability. In *Intelligent Virtual Agents* (2011), Springer, pp. 398{411.

[71] Neuron, P. Perception Neuron website, https://neuronmocap.com/, [Online; accessed 1-July-2020].

[72] Newell, A., Yang, K., and Deng, J. Stacked Hourglass Networks for Human Pose Estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2016), Springer, pp. 483{499.

[73] Oculus. Using deep neural networks for accurate hand-tracking on Oculus Quest, website, https://ai.facebook.com/blog/hand-tracking-deep-neural-networks, [Online; accessed 1-July-2020].

[74] Oikonomidis, I., Kyriazis, N., and Argyros, A. A. Tracking the articulated motion of two strongly interacting hands. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2012).

[75] OptiTrack. Active Components website, https://optitrack.com/products/active-components/, [Online; accessed 1-July-2020].

[76] Owlchemy Labs. Job Simulator website, https://jobsimulatorgame.com/, [Online; accessed 1-July-2020].

[77] Pavllo, D., Delahaye, M., Porssut, T., Herbelin, B., and Boulic, R. Real-time neural network prediction for handling two-hands mutual occlusions. *Computers & Graphics: X 2* (2019), 100011.

[78] Pavllo, D., Porssut, T., Herbelin, B., and Boulic, R. Real-time finger tracking using active motion capture: A neural network approach robust to occlusions. In *Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games* (New York, NY, USA, 2018), MIG '18, Association for Computing Machinery.

[79] PhaseSpace. PhaseSpace website, https://www.phasespace.com/, [Online; accessed 1-July-2020].

[80] Polhemus. website, https://polhemus.com/, [Online; accessed 1-July-2020].

[81] Pollard, N. S., and Zordan, V. Physically based grasping control from example. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2005), pp. 311{318.

[82] Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), IEEE, pp. 652{660.

[83] Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Advances in Neural Information Processing Systems* (2017), pp. 5099{5108.

[84] Qian, C., Sun, X., Wei, Y., Tang, X., and Sun, J. Realtime and robust hand tracking from depth. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2014), IEEE, pp. 1106{1113.

[85] Rashid, A., and Hasan, O. Wearable technologies for hand joints monitoring for rehabilitation: A survey. *Microelectronics Journal 88* (2019), 173{183.

[86] Ready At Dawn, Oculus Studios. Lone Echo website, https://www.echo.games/, [Online; accessed 1-July-2020].

[87] Riemer, M., Trojan, J., Beauchamp, M., and Fuchs, X. The rubber hand universe: On the impact of methodological differences in the rubber hand illusion. *Neuroscience & Biobehavioral Reviews 104* (2019), 268 { 280.

[88] Rijpkema, H., and Girard, M. Computer animation of knowledge-based human grasping. *ACM Siggraph Computer Graphics 25*, 4 (1991), 339{348.

[89] Robotics, D. Dexta Robotics website, https://www.dextarobotics.com/en-us/, [Online; accessed 1-July-2020].

[90] Rokoko. website, https://www.rokoko.com/en, [Online; accessed 1-July-2020].

[91] Romero, J., Kjellström, H., and Kragic, D. Hands in Action: Real-time 3D Reconstruction of Hands in Interaction with Objects. In *Proceedings of the International Conference on Robotics and Automation (ICRA)* (2010), IEEE, pp. 458{463.

[92] Romero, J., Tzionas, D., and Black, M. J. Embodied Hands: Modeling and Capturing Hands and Bodies Together. *ACM Transactions on Graphics (TOG) 36*, 6 (2017), 245:1{245:17.

[93] Rose III, C. F., Sloan, P.-P. J., and Cohen, M. F. Artist-directed inverse-kinematics using radial basis function interpolation. In *Computer Graphics Forum* (2001), vol. 20, Wiley Online Library, pp. 239{250.

[94] Sanchez-Vives, M. V., Spanlang, B., Frisoli, A., Bergamasco, M., and Slater, M. Virtual hand illusion induced by visuomotor correlations. *PLOS ONE 5*, 4 (2010), 1{6.

[95] Schröder, M., Maycock, J., and Botsch, M. Reduced marker layouts for optical motion capture of hands. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games* (2015), MIG.

[96] Schröder, M., Waltemate, T., Maycock, J., Röhlig, T., Ritter, H., and Botsch, M. Design and evaluation of reduced marker layouts for hand motion capture. *Computer Animation and Virtual Worlds 29*, 6 (2018), e1751.

[97] Schwind, V., Knierim, P., Chuang, L., and Henze, N. \Where's pinky?": The effects of a reduced number of fingers in virtual reality. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play* (New York, NY, USA, 2017), CHI PLAY '17, Association for Computing Machinery, pp. 507{{515.

[98] Schwind, V., Knierim, P., Tasci, C., Franczak, P., Haas, N., and Henze, N. \These are not my hands!": Effect of gender on the perception of avatar hands in virtual reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2017), CHI '17, Association for Computing Machinery, p. 1577{1582.

[99] Schwind, V., Lin, L., Di Luca, M., Jörg, S., and Hillis, J. Touch with foreign hands: The effect of virtual hand appearance on visual-haptic integration. In *Proceedings of the 15th ACM Symposium on Applied Perception* (New York, NY, USA, 2018), SAP '18, Association for Computing Machinery.

[100] Sense, S. Stretch Sense website, https://stretchsense.com/, [Online; accessed 1-July-2020].

[101] Slater, M., Perez-Marcos, D., Ehrsson, H. H., and Sanchez-Vives, M. V. Inducing illusory ownership of a virtual body. *Frontiers in Neuroscience 3*, 2 (2009), 214{220.

[102] Spurr, A., Song, J., Park, S., and Hilliges, O. Cross-Modal Deep Variational Hand Pose Estimation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), IEEE, pp. 89{98.

[103] Sridhar, S., Mueller, F., Oulasvirta, A., and Theobalt, C. Fast and Robust Hand Tracking Using Detection-Guided Optimization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), IEEE, pp. 3213{3221.

[104] Sridhar, S., Rhodin, H., Seidel, H.-P., Oulasvirta, A., and Theobalt, C. Real-time Hand Tracking Using a Sum of Anisotropic Gaussians Model. In *Proceedings of the International Conference on 3D Vision (3DV)* (2014), IEEE, pp. 319{326.

[105] Sturman, D. J., and Zeltzer, D. A survey of glove-based input. *IEEE Computer Graphics and Applications 14*, 1 (1994), 30{39.

[106] Supančič, J. S., Rogez, G., Yang, Y., Shotton, J., and Ramanan, D. Depth-based hand pose estimation: Methods, data, and challenges. *International Journal of Computer Vision (IJCV) 126*, 11 (2018), 1180{1198.

[107] Tagliasacchi, A., Schroeder, M., Tkach, A., Bouaziz, S., Botsch, M., and Pauly, M. Robust Articulated-ICP for Real-Time Hand Tracking. In *Computer Graphics Forum (Symposium on Geometry Processing)* (2015), vol. 34, Wiley Online Library, pp. 101{114.

[108] Talvas, A., Marchal, M., Duriez, C., and Otaduy, M. A. Aggregate constraints for virtual manipulation with soft fingers. *IEEE Transactions on Visualization and Computer Graphics 21*, 4 (2015), 452{461.

[109] Tan, D. J., Cashman, T., Taylor, J., Fitzgibbon, A., Tarlow, D., Khamis, S., Izadi, S., and Shotton, J. Fits Like a Glove: Rapid and Reliable Hand Shape Personalization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), IEEE, pp. 5610{5619.

[110] Tang, D., Jin Chang, H., Tejani, A., and Kim, T.-K. Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2014), IEEE, pp. 3786{3793.

[111] Tang, D., Taylor, J., Kohli, P., Keskin, C., Kim, T.-K., and Shotton, J. Opening the Black Box: Hierarchical Sampling Optimization for Estimating Human Hand Pose. In *Proceedings of the International Conference on Computer Vision (ICCV)* (2015), IEEE, pp. 3325{3333.

[112] Taylor, J., Bordeaux, L., Cashman, T., Corish, B., Keskin, C., Sharp, T., Soto, E., Sweeney, D., Valentin, J., Luff, B., Topalian, A., Wood, E., Khamis, S., Kohli, P., Izadi, S., Banks, R., Fitzgibbon, A., and Shotton, J. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM Trans. Graph. 35*, 4 (July 2016), 143:1{143:12.

[113] Taylor, J., Tankovich, V., Tang, D., Keskin, C., Kim, D., Davidson, P., Kowdle, A., and Izadi, S. Articulated distance fields for ultra-fast tracking of hands interacting. *ACM Trans. Graph. 36*, 6 (Nov. 2017), 244:1{244:12.

[114] Tian, H., Wang, C., Manocha, D., and Zhang, X. Realtime hand-object interaction using learned grasp space for virtual environments. *IEEE Transactions on Visualization and Computer Graphics 25*, 8 (2019), 2623{2635.

[115] Tkach, A., Pauly, M., and Tagliasacchi, A. Sphere-meshes for real-time hand modeling and tracking. *ACM Trans. Graph. 35*, 6 (Nov. 2016), 222:1{222:11.

[116] Tompson, J., Stein, M., Lecun, Y., and Perlin, K. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Trans. Graph. 33*, 5 (Aug. 2014), 169:1{169:10.

[117] Tsakiris, M., and Haggard, P. The rubber hand illusion revisited: visuotactile integration and self-attribution. *Journal of Experimental Psychology: Human Perception and Performance 31*, 1 (feb 2005), 80{91.

[118] Tzionas, D., Ballan, L., Srikantha, A., Aponte, P., Pollefeys, M., and Gall, J. Capturing Hands in Action using Discriminative Salient Points and Physics Simulation. *International Journal of Computer Vision (IJCV) 118*, 2 (2016), 172{193.

[119] Verschoor, M., Lobo, D., and Otaduy, M. A. Soft hand simulation for smooth and robust natural interaction. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2018), IEEE, pp. 183{190.

[120] Vicon. Pulsar website, https://www.vicon.com/hardware/devices/pulsar/, [Online; accessed 1-July-2020].

[121] Vignais, N., Cocchiarella, D., Kociolek, A., and Keir, P. Dynamic assessment of finger joint loads using kinetic and kinematic measurements. In *Digital Human Modeling Symposium* (2013).

[122] Vocelle, A. R., Shafer, G., and Bush, T. R. Complex thumb motions and their potential clinical value in identifying early changes in function. *Clinical Biomechanics 73* (2020), 63{70.

[123] Wan, C., Yao, A., and Van Gool, L. Hand pose estimation from local surface normals. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2016), Springer, pp. 554{569.

[124] Wang, B., Matcuk, G., and Barbič, J. Hand modeling and simulation using stabilized magnetic resonance imaging. *ACM Trans. Graph. 38*, 4 (July 2019).

[125] Wang, J., Mueller, F., Bernard, F., and Theobalt, C. Generative Model-Based Loss to the Rescue: A Method to Overcome Annotation Errors for Depth-Based Hand Pose Estimation. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition (FG)* (2020), IEEE, pp. 93{100.

[126] Wang, R. Y., and Popović, J. Real-time hand-tracking with a color glove. *ACM transactions on graphics (TOG) 28*, 3 (2009), 1{8.

[127] Wang, Y., and Neff, M. Data-driven glove calibration for hand motion capture. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2013), ACM, pp. 15{24.

[128] Wang, Y., Tree, J. E. F., Walker, M., and Neff, M. Assessing the impact of hand motion on virtual character personality. *ACM Transactions on Applied Perception (TAP) 13*, 2 (2016), 1{23.

[129] Wheatland, N., Jörg, S., and Zordan, V. Automatic hand-over animation using principle component analysis. In *Proceedings of Motion on Games* (New York, NY, USA, 2013), MIG '13, Association for Computing Machinery, p. 197{202.

[130] Wheatland, N., Wang, Y., Song, H., Neff, M., Zordan, V., and Jörg, S. State of the art in hand and finger modeling and animation. *Computer Graphics Forum 34*, 2 (2015), 735{760.

[131] Xiang, D., Joo, H., and Sheikh, Y. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 10965{10974.

[132] Ye, Y., and Liu, C. K. Synthesis of detailed hand manipulations using contact sampling. *ACM Transactions on Graphics (TOG) 31*, 4 (2012), 1{10.

[133] Yuan, S., Garcia-Hernando, G., Stenger, B., Moon, G., Yong Chang, J., Mu Lee, K., Molchanov, P., Kautz, J., Honari, S., Ge, L., Yuan, J., Chen, X., Wang, G., Yang, F., Akiyama, K., Wu, Y., Wan, Q., Madadi, M., Escalera, S., Li, S., Lee, D., Oikonomidis, I., Argyros, A., and Kim, T.-K. Depth-Based 3D Hand Pose Estimation: From Current Achievements to Future Goals. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), IEEE, pp. 2636{2645.

[134] Yuan, Y., and Steed, A. Is the rubber hand illusion induced by immersive virtual reality? In *2010 IEEE Virtual Reality Conference (VR)* (March 2010), pp. 95{102.

[135] Zhang, J., and Hommel, B. Body ownership and response to threat. *Psychological Research* (2015), 1{10.

[136] Zhao, J., and Badler, N. I. Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Transactions on Graphics (TOG) 13*, 4 (1994), 313{336.

[137] Zhao, W., Zhang, J., Min, J., and Chai, J. Robust realtime physics-based motion control for human grasping. *ACM Transactions on Graphics (TOG) 32*, 6 (2013), 1{12.

[138] Zhu, Y., Ramakrishnan, A. S., Hamann, B., and Neff, M. A system for automatic animation of piano performances. *Computer Animation and Virtual Worlds 24*, 5 (2013), 445{457.

[139] Zimmermann, C., and Brox, T. Learning to Estimate 3D Hand Pose from Single RGB Images. In *Proceedings of the International Conference on Computer Vision (ICCV)* (2017), IEEE, pp. 4903{4911.

SIGGRAPH THINK BEYOND

2020 S2020.SIGGRAPH.ORG

VIRTUAL HANDS IN VR

Motion Capture, Synthesis, and Perception

SIGGRAPH THINK BEYOND

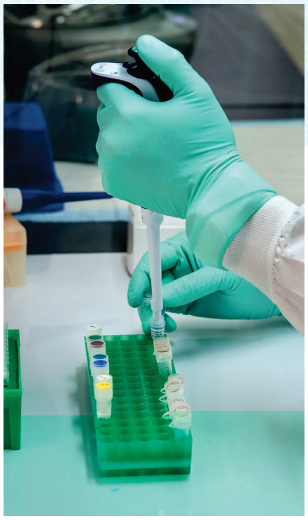
2020 S2020.SIGGRAPH.ORG

INTRODUCTION

Sophie Jörg

WELCOME AND OVERVIEW

Sophie Jörg



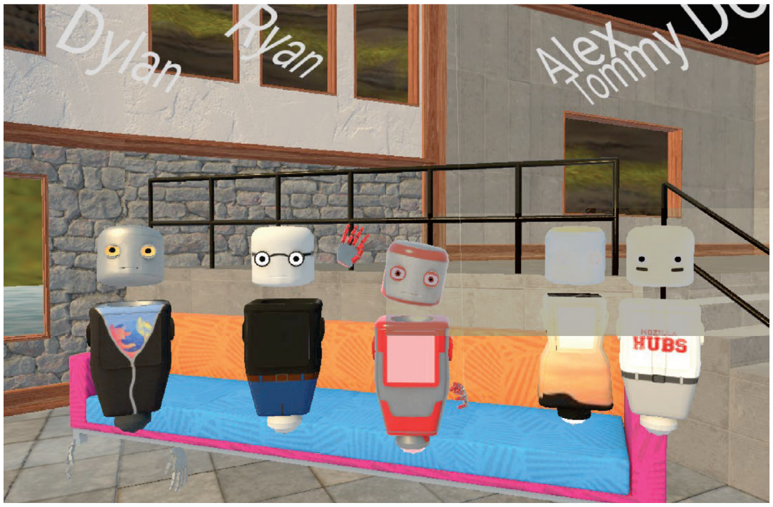
SELECTION OF VR MEETING SPACES AND HARDWARE AVAILABLE

Example products:

VR meeting spaces:
VRChat, Mozilla Hubs, AltspaceVR,
Engage, Facebook Horizon

HMDs: HTC Vive Cosmos, Oculus Rift S,
Sony PlayStation VR, Valve Index

Hand/Finger Tracking: Oculus Quest, Leap



Hubs by Mozilla

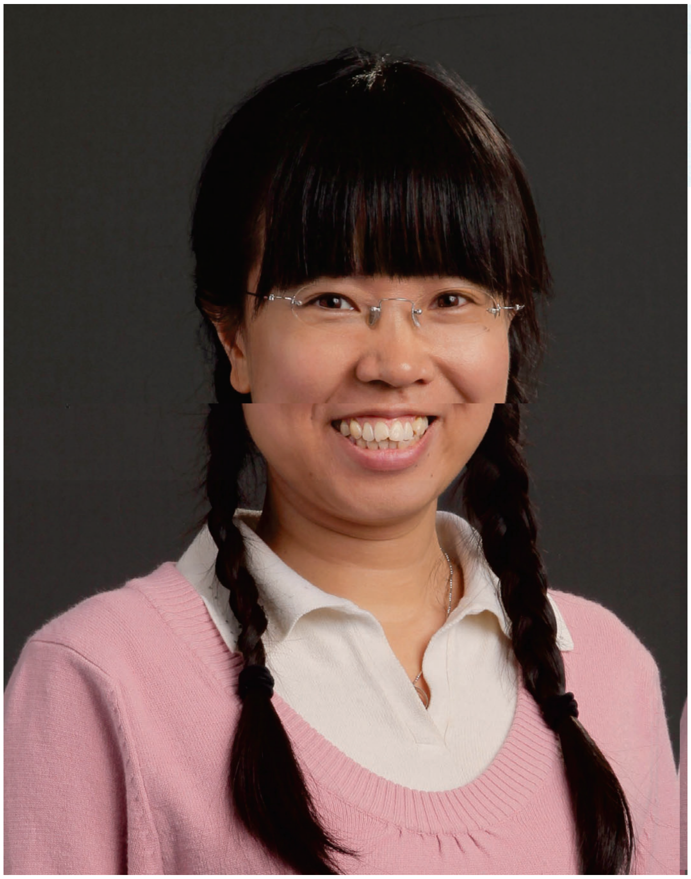
COURSE OVERVIEW

- **Introduction** (15 Minutes)
  - **Welcome and Overview** (5 minutes)
  - **Introduction to Virtual Hands** (10 minutes)
- **Motion Capturing Hands** (75 Minutes)
  - **Optical Marker-Based Motion Capture** (25 Minutes)
  - **Gloves and Non-Optical Approaches** (10 Minutes)
  - **Image- and Depth-Sensor-Based Methods** (40 Minutes)
- **Break** (5 Minutes)
- **Hand Motion Synthesis** (45 Minutes)
  - **Kinematic Hand Motion Synthesis** (15 Minutes)
  - **Physical Modeling** (30 Minutes)
- **Perception of Virtual Hands** (25 Minutes)
- **Conclusions and Q&A** (15 Minutes)

SOPHIE JÖRG

CLEMSON UNIVERSITY
Associate Professor, School of Computing
sjoerg@clemson.edu
https://people.cs.clemson.edu/ ~ sjoerg/

Research Interests: Character Animation, Motion Capture, Data-driven Algorithms, Human Perception, Virtual Reality, Hand Motions, Human-Computer Interaction



YUTING YE

FACEBOOK REALITY LABS
Research Scientist
yuting.ye@fb.com
http://yutingye.info

Research Interests: Physics-based Simulation and Control, Hand Manipulation, Motion Planning, Motion Retargeting, Optimal Control Algorithms, Performance Capture, Machine Learning

MICHAEL NEFF

UNIVERSITY OF CALIFORNIA, DAVIS

Department of Computer Science and Department of Cinema and Digital Media, University of California, Davis

mpneff@ucdavis.edu

https://www.cs.ucdavis.edu/~neff/

Research Interests: Character Animation, Gesture and Nonverbal Communication, Applying Concepts from the Performing Arts to Animation Tools, Physics-based Models, Interactive Animation

FRANZISKA MUELLER

MAX PLANCK INSTITUTE FOR INFORMATICS

PhD Student, Graphics, Vision and Video Group

frmueller@mpi-inf.mpg.de

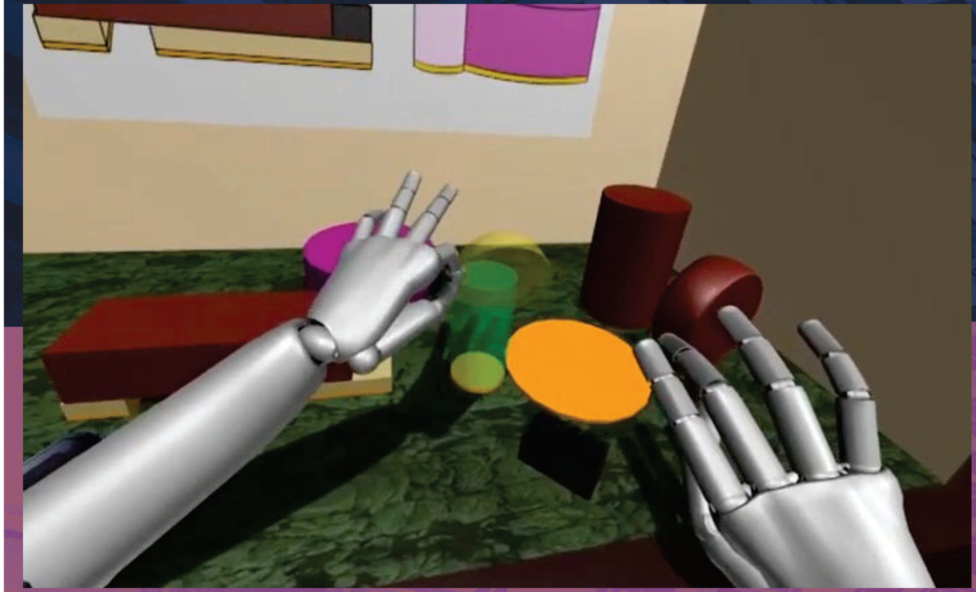
https://people.mpi-inf.mpg.de/~frmueller/

Research Interests: Real-time Tracking, Machine Learning, Computer Vision, Model- and Optimization-based Reconstruction of Articulated Motion

VICTOR ZORDAN

CLEMSON UNIVERSITY
School of Computing, Clemson University
vbz@clemson.edu
https://people.cs.clemson.edu/~vbz/

Research Interests: Physical Simulation, Motion Capture, Animation, Games, Medical and Training Applications, Virtual Worlds, Fabrication



INTRODUCTION TO VIRTUAL HANDS

Sophie Jörg

JOB SIMULATOR

Selected hand poses are displayed based on controller input

Graspable objects are highlighted

The hand disappears when the object is grasped

The object then follows the invisible hand

Vibration gives feedback to user when an object is grasped or released

FURTHER EXAMPLES

In some games, the hand pose adjusts to the environment.

Examples: *The Climb* (Crytek), *Lone Echo* (Ready at Dawn)

More and more games appear with directly tracked hand motions,

e.g, using the Leap Motion or Oculus Quest

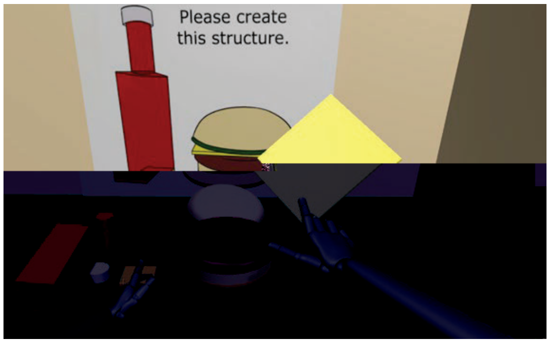
One can, for example, play a virtual piano

CHALLENGES

- Accurate hand tracking
  - High number of degrees of freedom (DOF)
  - Different in scale from the body (smaller scale)
  - Self-occlusions
- Human perception
- Responsive grasp and release recognition
- Haptic feedback
- Lack of solid surfaces that would prevent intersections

USING CONTROLLERS VS. TRACKING HANDS

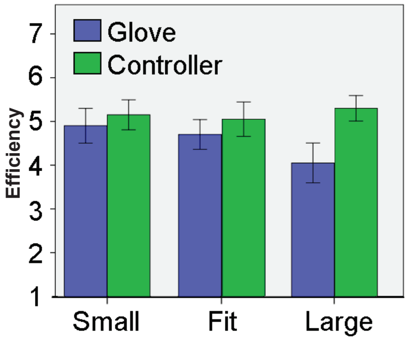
Experiment comparing the usage of gloves and controllers for a game-like pick and place task.

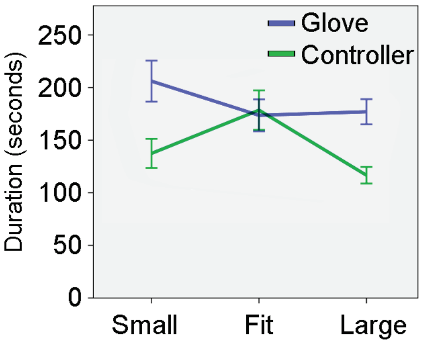


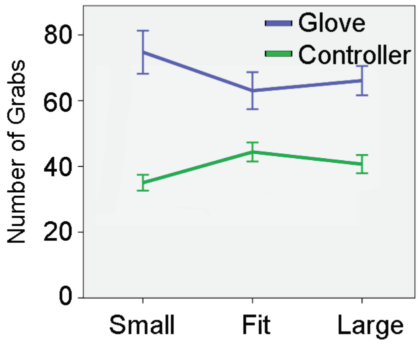
***The Effect of Hand Size and Interaction Modality on the Virtual Hand Illusion*. Lorraine Lin, Aline Normovle, Alexandra Adkins, Yu Sun, Andrew Robb, Yuting Ye, Massimiliano Di Luca, and Sophie Jörg. IEEE VR 2018.**

USING CONTROLLERS VS. TRACKING HANDS

Controller felt more efficient, task duration was shorter and the number of grabs and of drops was lower

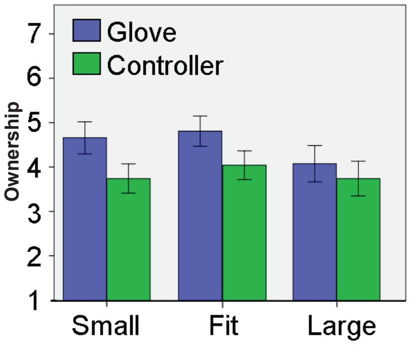


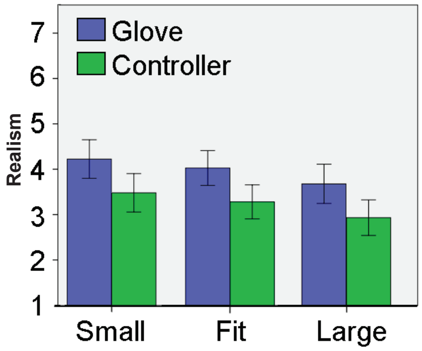




USING CONTROLLERS VS. TRACKING HANDS

Ownership was rated higher when using the gloves, and the virtual hands looked more realistic to the participants





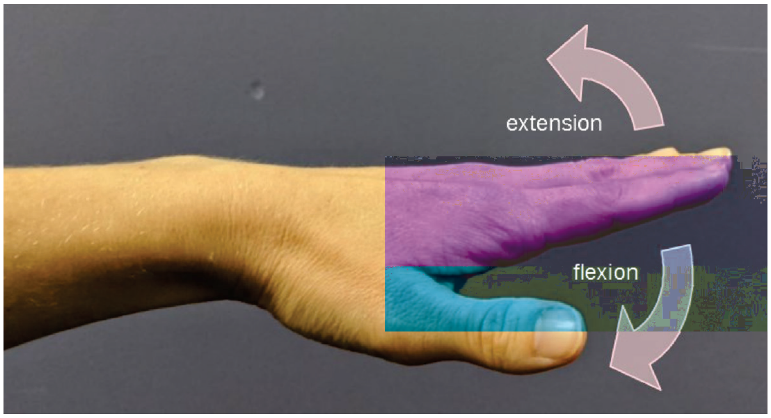
USING CONTROLLERS VS. TRACKING HANDS

When asked for their preferences:

Glove	**13**	“easier to control,” “it felt more realistic,” “more immersive,” “more fun,” “more comfortable,” “I prefer the gloves since I was able to move all of my fingers and it looked just like my own hands,” “I felt like I was [going to] drop the controllers because I had to keep thinking ‘I’m using controllers, I can’t let go of these.” 6 participants who preferred the Glove condition reported that they felt the Controller condition had better feedback.
Controller	**6**	“it was more precise when I was picking things up,” “more responsive,” “the gloves were more immersive, but the controllers seemed to work better,” “with the gloves there wasn’t any real feedback.” 1 participant would prefer the gloves if they “worked like my real hands.”
No preference	**1**	

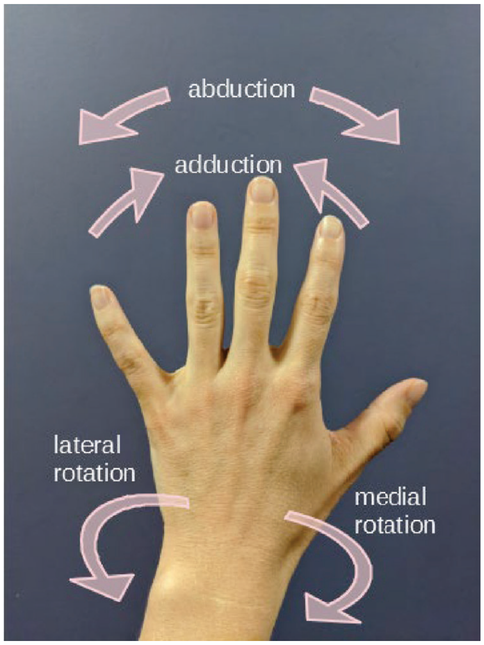
HOW THE HANDS MOVE

- Flexion: Bending in the anterior direction (making a fist)
- Extension: Straightening or bending in the posterior direction

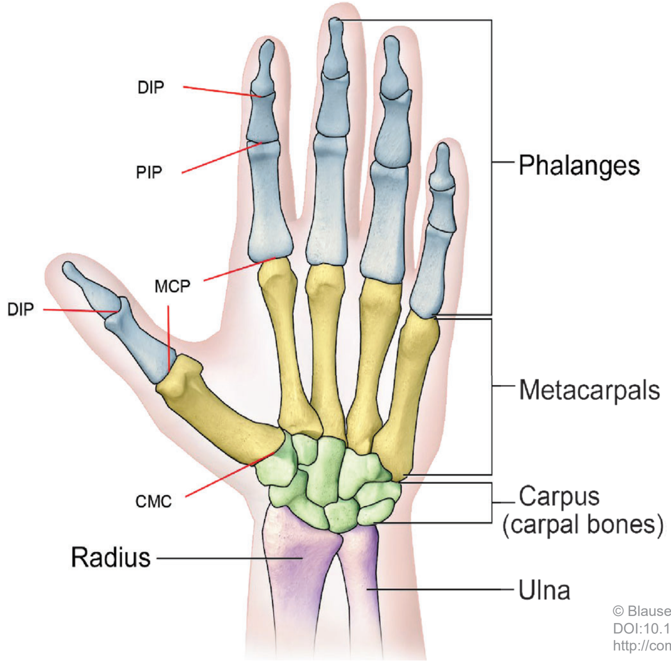


HOW THE HANDS MOVE

- Abduction: Movement away from the center of the body (fingers spread)
- Adduction: Movement toward the center of the body (fingers together)
- Medial/Lateral Rotation



BONES OF THE HAND



- 27 Bones
  - Carpus
  - Metacarpals
  - Phalanges

- Joints
  - CMC: Carpometacarpal joint
  - MCP: Metacarpophalangeal joint
  - PIP: Proximal interphalangeal joint
  - DIP: Distal interphalangeal joint

© Blausen.com staff (2014). "Medical gallery of Blausen Medical 2014". WikiJournal of Medicine 1 (2). DOI:10.15347/wjm/2014.010. ISSN 2002-4436.
http://commons.wikimedia.org/wiki/file:blausen_0440_handbones.png, edits made for this publication

HAND MODEL REPRESENTATIONS

- Fully anatomical model is complicated to replicate and computationally expensive for many applications

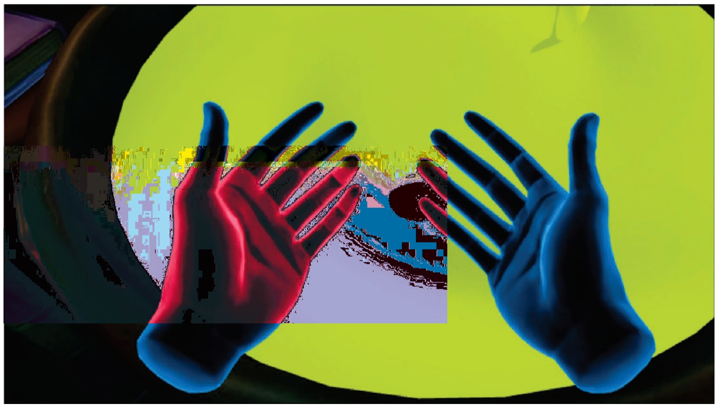
- Simplifications often made, supported by anatomy
  - Bones represented using rigid bodies
  - Reduced number of bones or joints, e.g., some wrist bones/articulations considered negligible
  - Joints represented with fewer DOFs

- Personalized skeletons possible
  - Very accurate motions possible
  - Centers/axes of rotation can be determined for each joint using optical motion capture or medical imaging

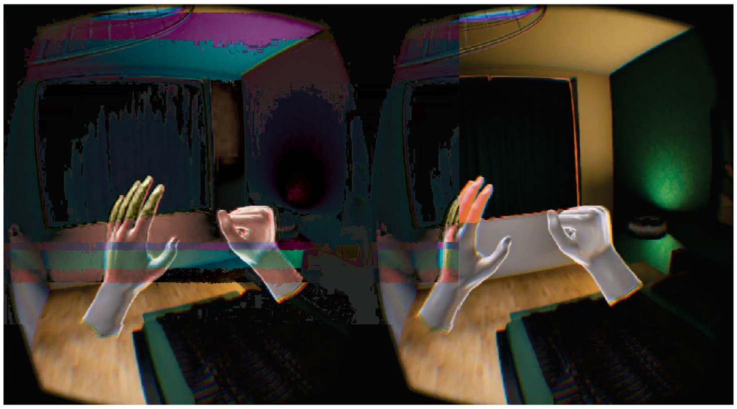
MOTION CAPTURING FINGERS

Yuting Ye Michael Neff Franziska Mueller

APPLICATIONS IN THE VR MARKET



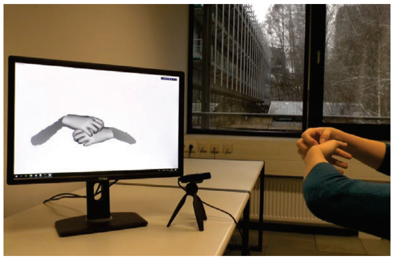
Elixir on Oculus Quest
© Magnopus © Facebook Reality Labs

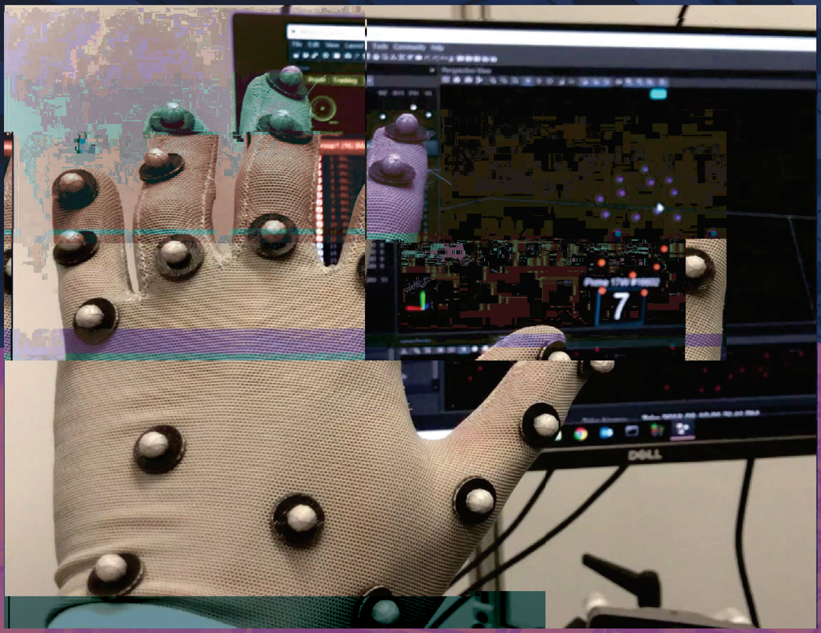


ArchVis on Oculus Rift + Leap Motion
© matburr@Youtube, CC BY

TECHNOLOGIES FOR FINGER MOTION CAPTURE

- Optical markers
- Gloves and non-optical sensors
- Images and depth sensors





MOTION CAPTURING FINGERS: OPTICAL MARKER-BASED APPROACHES

Yuting Ye

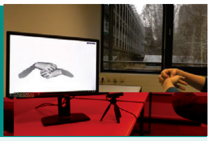
TECHNOLOGIES FOR FINGER MOTION CAPTURE

Markers

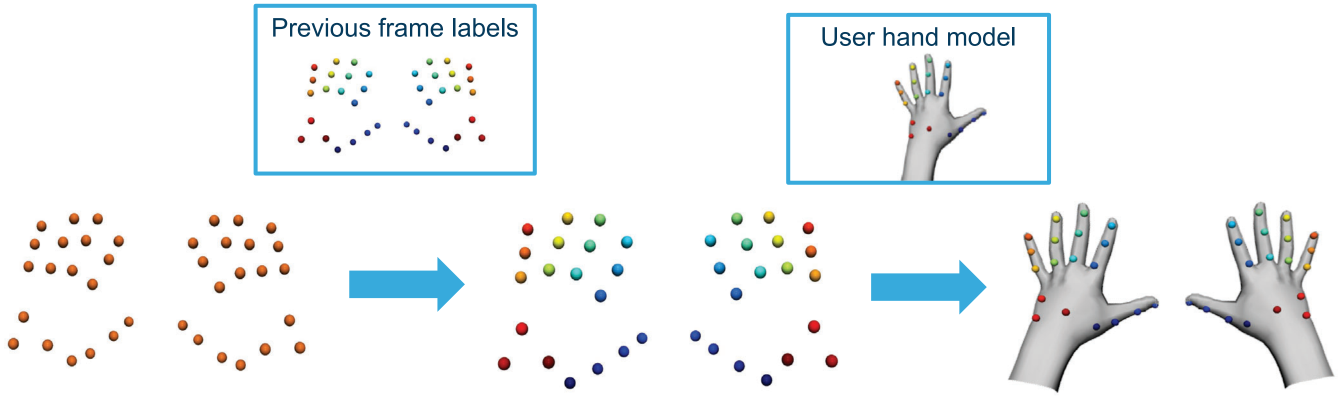
- Accurate world space positional tracking
- High framerate
- Relatively easy to use with mature solutions
- Require a dedicated instrumented space
- Labeling and occlusion
- Costs $$$

Gloves

Images



MOTION CAPTURE PIPELINE



3D marker reconstruction

Missing markers due to occlusion or poor visibility

Marker labeling

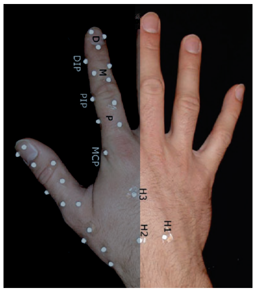
Wrong labels and tracking loss

Hand pose fitting

Mismatched hand model and insufficient degrees of freedom

MARKER LAYOUTS

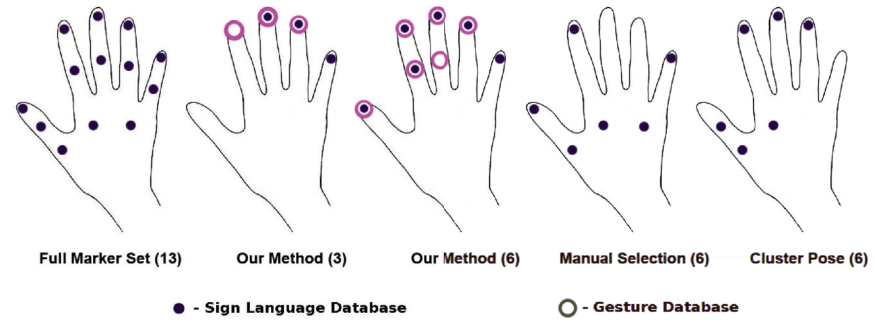
[Alexanderson et al. 2017]
© Alexanderson et al.



[Chang and Pollard 2007]
© Elsevier

- Marker size and shape
  - Visibility
  - Density
- Attachment methods
  - Glue
  - Velcro
  - Glove
- Marker placement
  - Freedom of movement
  - Easy to recover poses

REDUCED MARKER LAYOUT AND ANIMATION

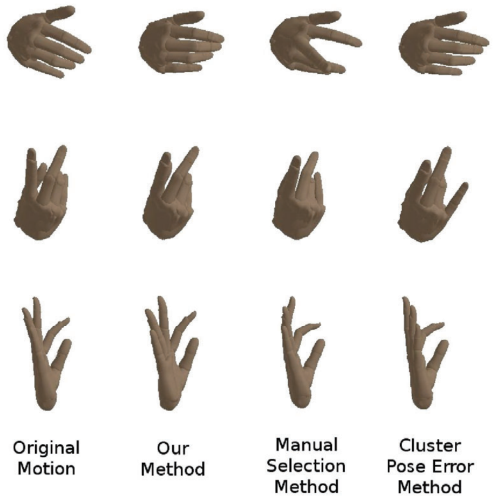


Automatic Hand-Over Animation using Principle Component Analysis.
Nkenge Wheatland, Sophie Joerg, Victor Zordan. Motion In Games 2013

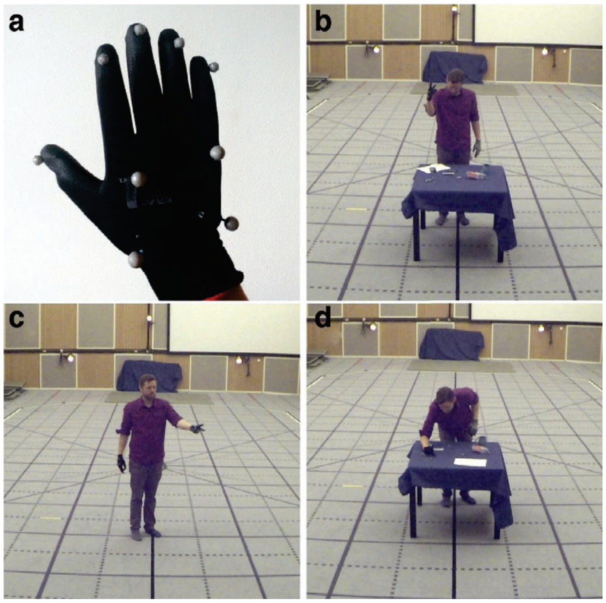
- Apply PCA to all markers in a dataset and pick the important ones
- Run regression from reduced marker set positions to joint angle PCs
- Recover full joint angles from regressed PCs

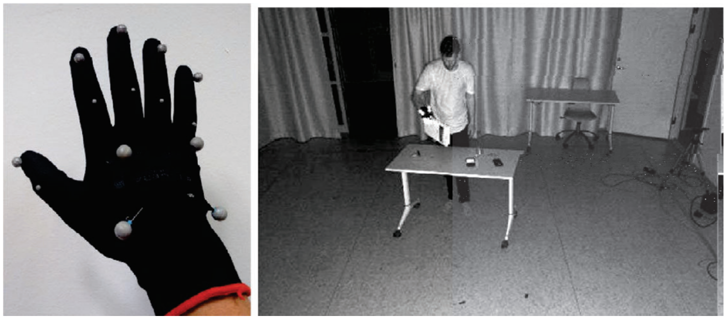
REDUCED MARKER LAYOUT AND ANIMATION

Original Motion
Six Markers



LABELING REDUCED MARKER SETS

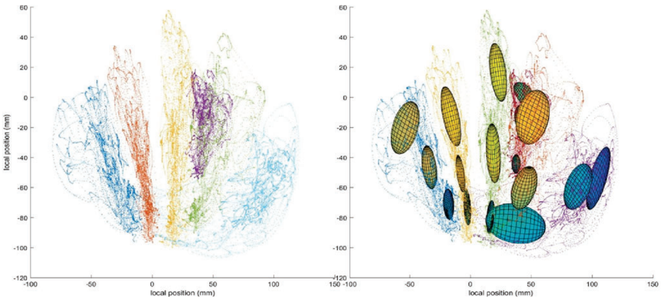




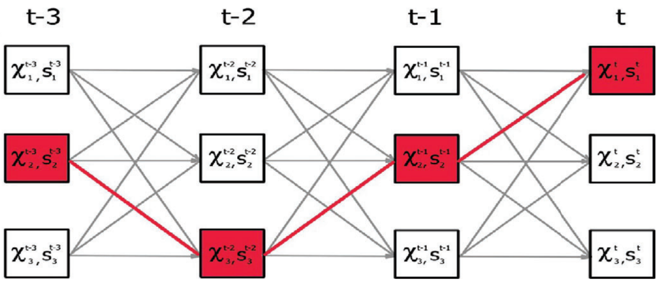
***Real-time labeling of non-rigid motion capture marker sets*.**
Simon Alexanderson, Carol O'Sullivan, and Jonas Beskow.
Computers & Graphics 2017. © Alexanderson et al.

LABELING REDUCED MARKER SETS

Fingertip positions

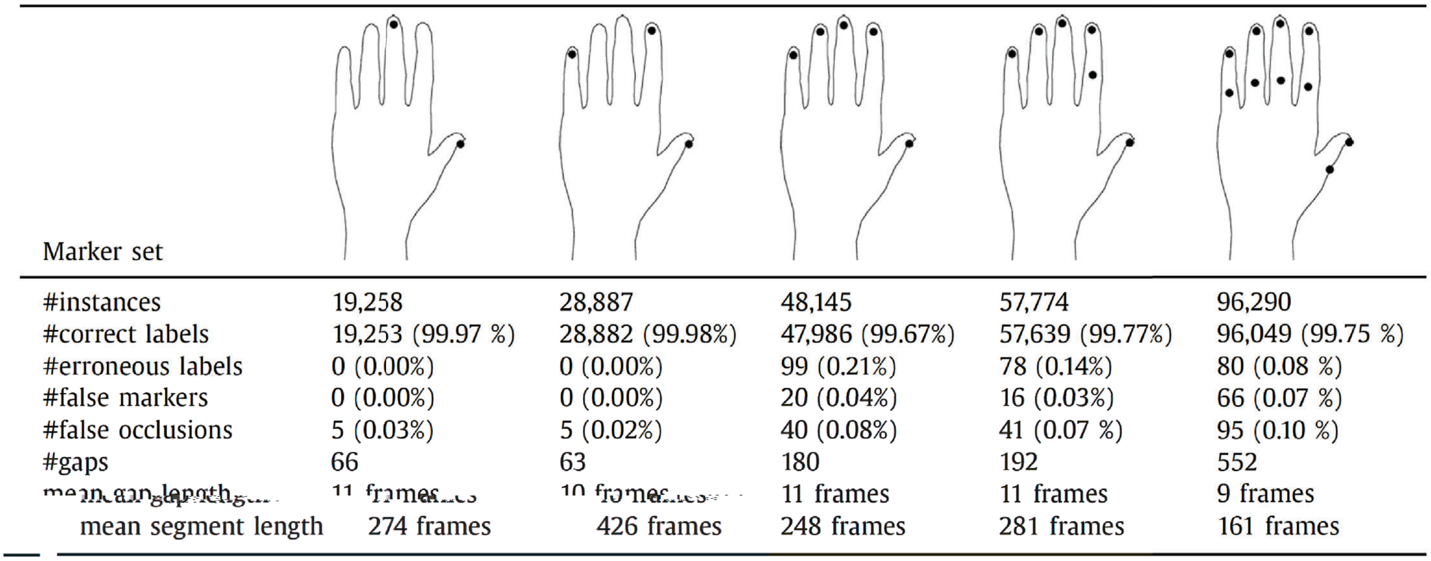


Viterbi algorithm


t-3
t-2
t-1
t
χ^{t-3}_1, s^{t-3}_1
χ^{t-3}_2, s^{t-3}_2
χ^{t-3}_3, s^{t-3}_3
χ^{t-2}_1, s^{t-2}_1
χ^{t-2}_2, s^{t-2}_2
χ^{t-2}_3, s^{t-2}_3
χ^{t-1}_1, s^{t-1}_1
χ^{t-1}_2, s^{t-1}_2
χ^{t-1}_3, s^{t-1}_3
χ^{t}_1, s^{t}_1
χ^{t}_2, s^{t}_2
χ^{t}_3, s^{t}_3

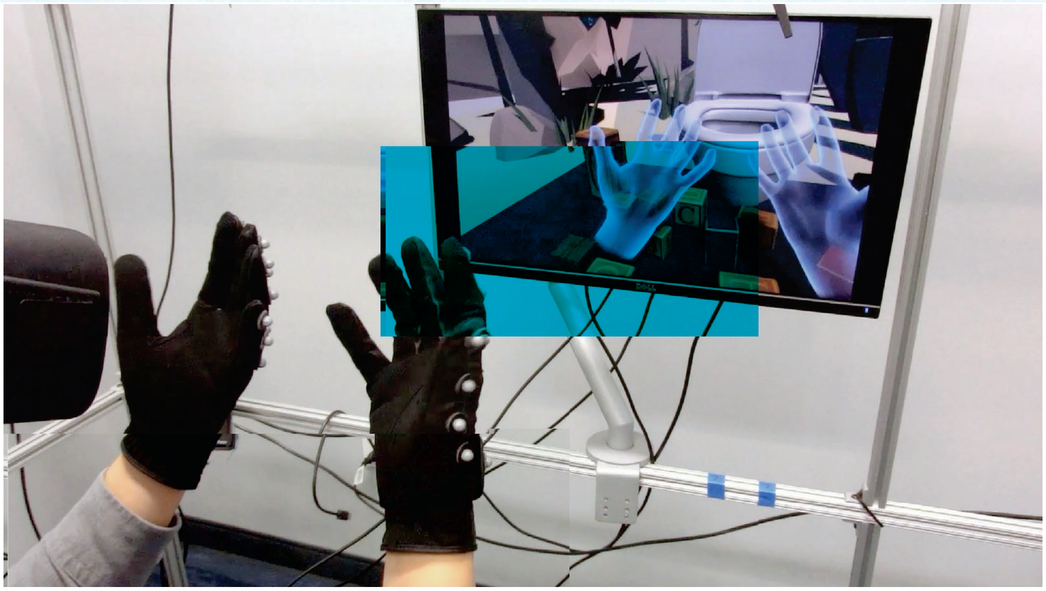
1. Apply Kalman filter to maker positions from the previous frame
2. Compute current frame marker position probability based on GMM
3. Apply Viterbi algorithm to obtain most likely labels from history

LABELING REDUCED MARKER SETS



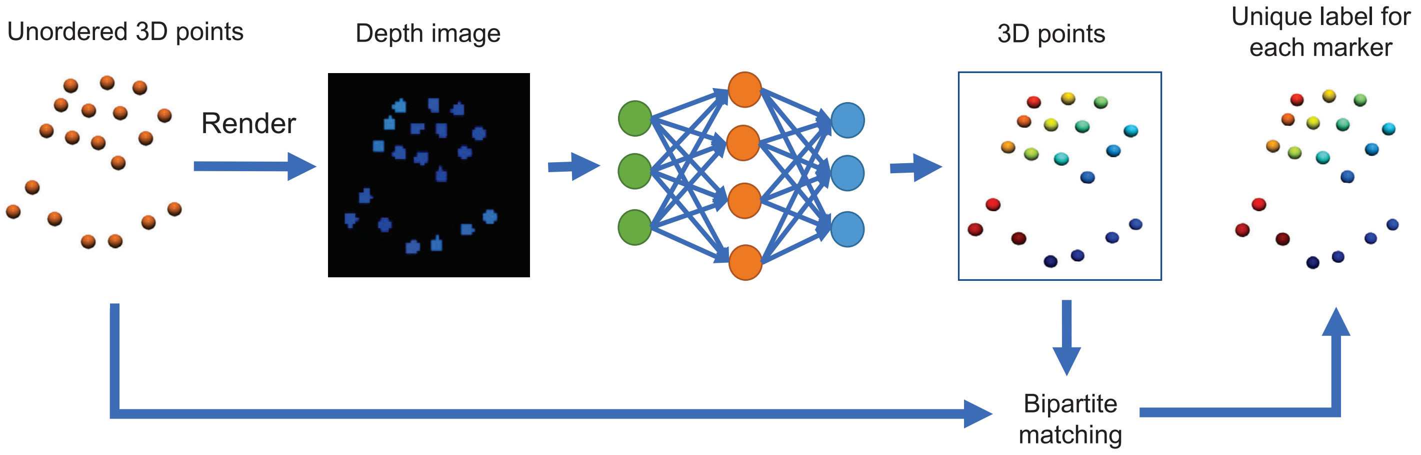
Marker set					
#instances	19,258	28,887	48,145	57,774	96,290
#correct labels	19,253 (99.97 %)	28,882 (99.98%)	47,986 (99.67%)	57,639 (99.77%)	96,049 (99.75 %)
#erroneous labels	0 (0.00%)	0 (0.00%)	99 (0.21%)	78 (0.14%)	80 (0.08 %)
#false markers	0 (0.00%)	0 (0.00%)	20 (0.04%)	16 (0.03%)	66 (0.07 %)
#false occlusions	5 (0.03%)	5 (0.02%)	40 (0.08%)	41 (0.07 %)	95 (0.10 %)
#gaps	66	63	180	192	552
mean gap length	11 frames	10 frames	11 frames	11 frames	9 frames
mean segment length	274 frames	426 frames	248 frames	281 frames	161 frames

LABELING A FULL MARKER SET



Online optical marker-based hand tracking with deep labels.
Shangchen Han, Beibei Liu, Rob Wang, Yuting Ye, Chris Twigg, Kenrick Kin. SIGGRAPH 2018

LABELING AS DEEP IMAGE REGRESSION



Unordered 3D points

Render

Depth image

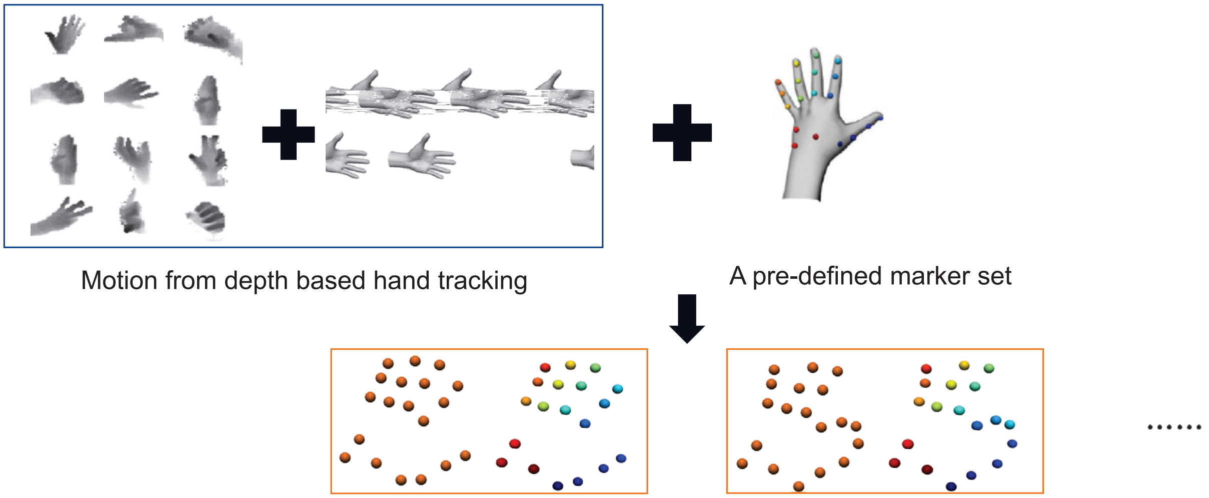
3D points

Unique label for each marker

Bipartite matching

SYNTHETIC TRAINING DATA

Real hand motion + synthetic marker positions



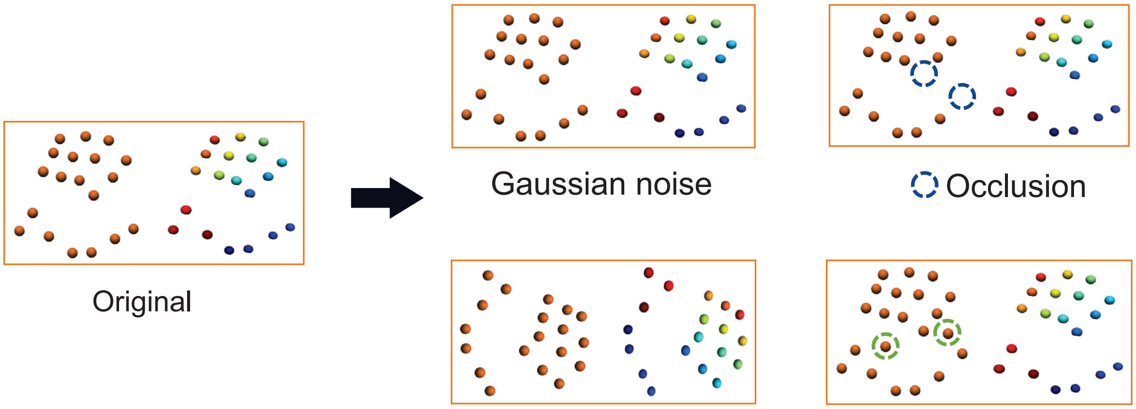
Motion from depth based hand tracking

A pre-defined marker set

"Spread"

"Pinch"

DATA AUGMENTATION



Original

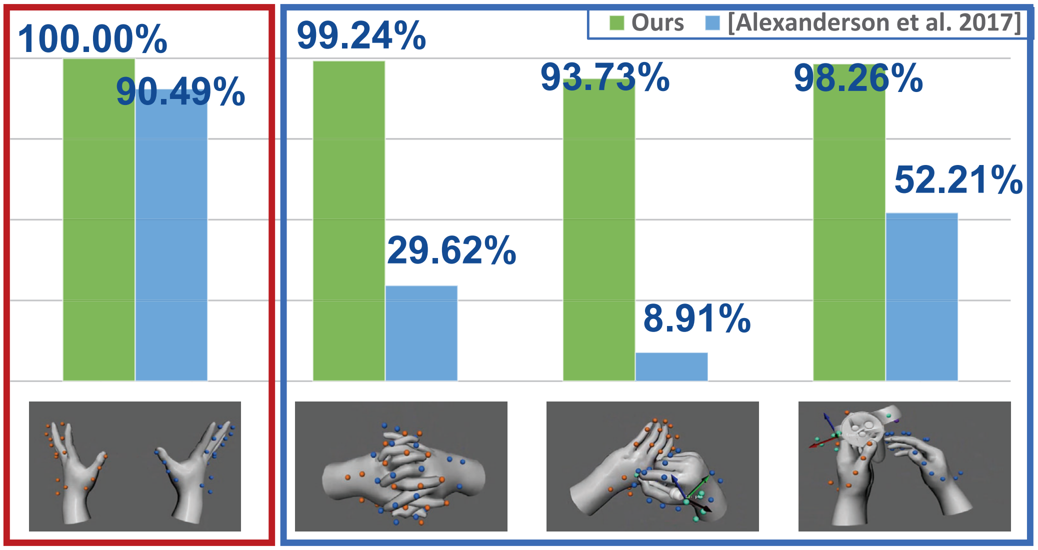
Gaussian noise

◌ Occlusion

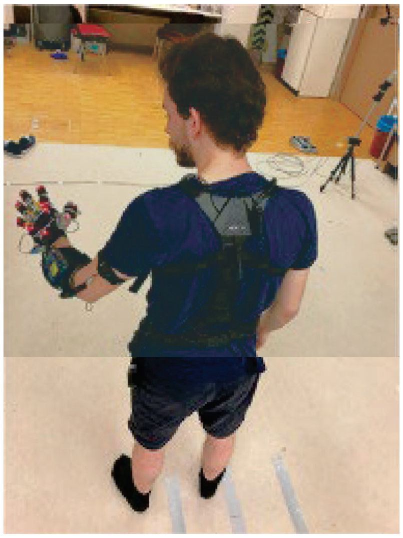
Random camera view

◌ Ghost marker

LABELING A FULL MARKER SET

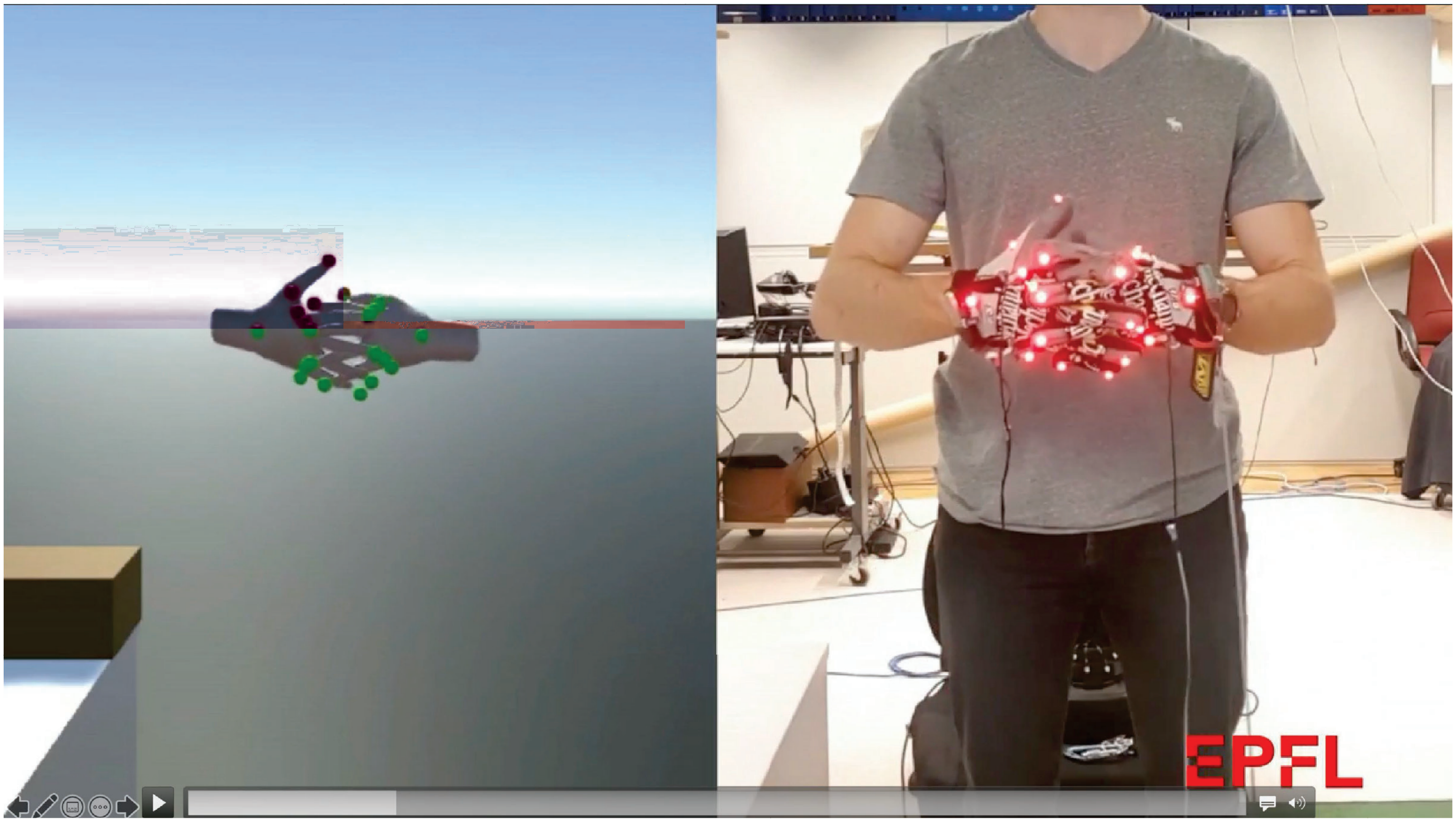


ACTIVE MARKERS



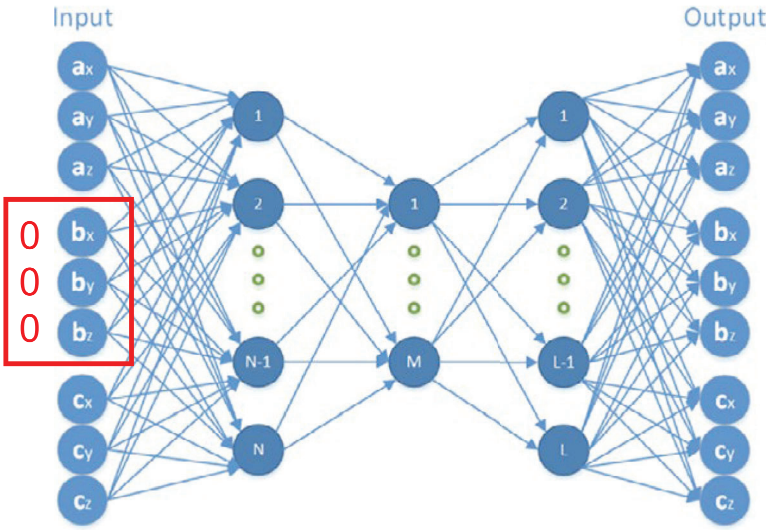
- LED markers emit light rather than reflect light
- Self-identifying for automatic labeling
- Larger capture volume
- More complex setup: wires, power, sync
- (More) prone to occlusion

Real-time neural network prediction for handling two-hands mutual occlusions.
D. Pavllo, M. Delahaye, T. Porssut, B. Herbelin, R. Boulic, Computers & Graphics 2019 ©Pavllo et al.



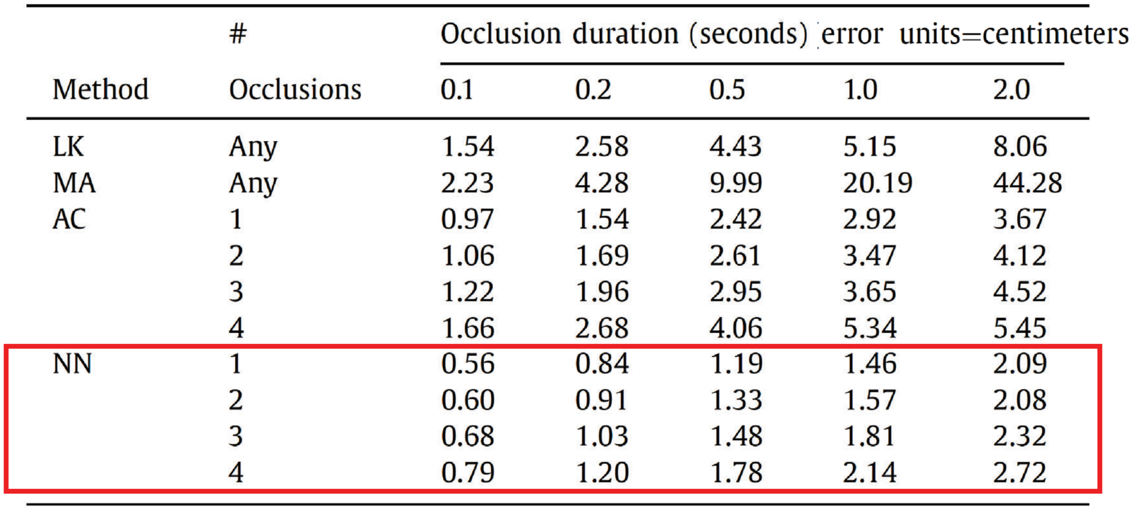
HANDLE MARKER OCCLUSION

SIGGRAPH THINK BEYOND 2020 S2020.SIGGRAPH.ORG



- Autoencoder architecture
- Randomly dropout input markers in training
- Blend occluded markers in and out to reduce discontinuity

HANDLE MARKER OCCLUSION



	#	Occlusion duration (seconds) error units=centimeters				
Method	Occlusions	0.1	0.2	0.5	1.0	2.0
LK	Any	1.54	2.58	4.43	5.15	8.06
MA	Any	2.23	4.28	9.99	20.19	44.28
AC	1	0.97	1.54	2.42	2.92	3.67
	2	1.06	1.69	2.61	3.47	4.12
	3	1.22	1.96	2.95	3.65	4.52
	4	1.66	2.68	4.06	5.34	5.45
NN	1	0.56	0.84	1.19	1.46	2.09
	2	0.60	0.91	1.33	1.57	2.08
	3	0.68	1.03	1.48	1.81	2.32
	4	0.79	1.20	1.78	2.14	2.72

LK last known position, **MA** moving average, **AC** affine combinations, **NN** neural network

REFERENCES

- Real-time neural network prediction for handling two-hands mutual occlusions, Pavllo et al., Computers & Graphics 2019
- Online optical marker-based hand tracking with deep labels, Han et al., SIGGRAPH 2018
- Design and evaluation of reduced marker layouts for hand motion capture, Schröder et al., Computer Animation & Virtual Worlds 2018
- Real-time labeling of non-rigid motion capture marker sets, Alexanderson et al., Computers & Graphics 2017
- State of the art in hand and finger modeling and animation, Wheatland et al., Eurographics STAR 2015
- Automatic hand-over animation using principle component analysis, Wheatland et al., Motion In Games 2013
- Sleight of hand: Perception of finger motion from reduced marker sets, Hoyet et al., I3D 2012
- Automatic hand-over animation for free-hand motions from low resolution input, Kang et al., Motion In Games 2012
- Feature selection for grasp recognition from optical markers, Chang et al., IROS 2007

APPLICATIONS

- Applications in VR
  - Prototyping and user studies
  - Commercial VR experiences (eg. VR Arcade)
  - Digital avatars

- Applications in research
  - Combined body and hands capture in real time
  - Complex hand-object interaction capture
  - High quality hand motions for deep learning
  - Synthetic data for image based deep learning

MOTION CAPTURING FINGERS: GLOVES AND NON-OPTICAL APPROACHES

Michael Neff

OPTICAL ALTERNATIVES

- All optical techniques suffer from potential occlusions
- Gloves allow sensors to be attached to the hands
  - Non-optical techniques avoid occlusions
  - Require setup

[CC0, Max Pixel]

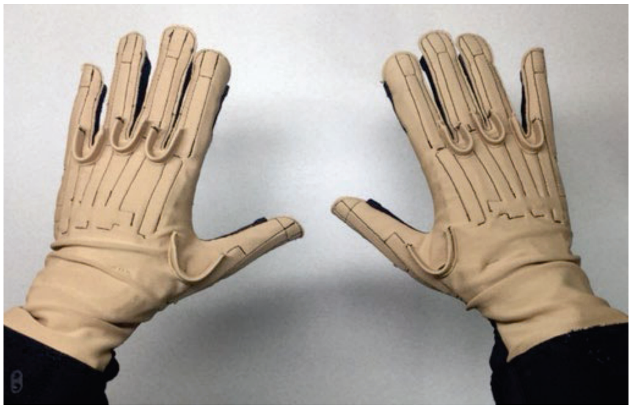
TECHNOLOGIES

BEND SENSORS

- How they work:
  - Resistance changes as physical sensors bend
    - Creates variation in electric signal
  - Aim for linear relationship between bend and signal
  - Attach sensors to specific DOFs using gloves

CyberGlove II

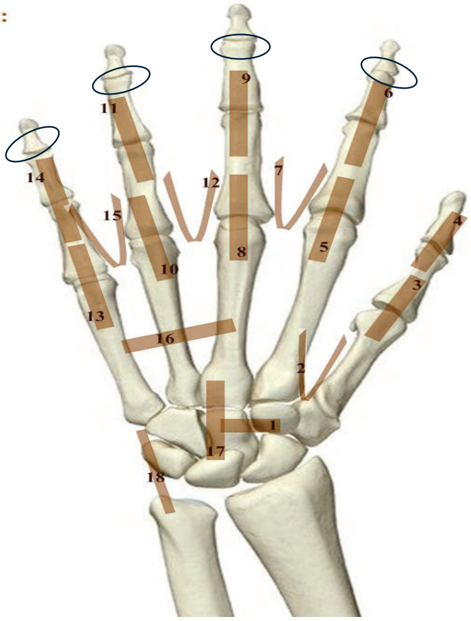
GLOVE SENSOR & KINEMATIC HAND MODEL



Glove attaches physical sensors to the hand.

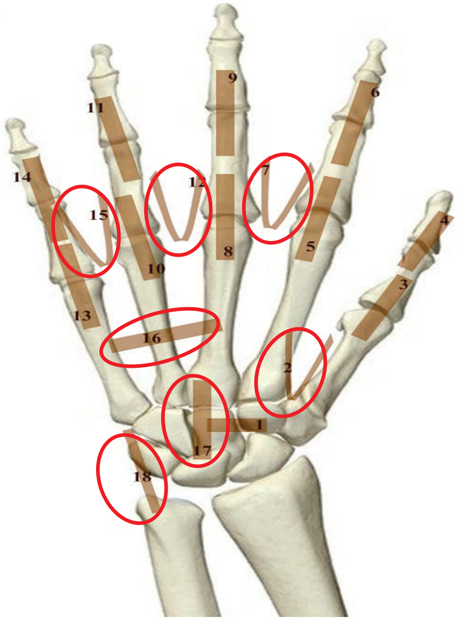
Sensor Layout:

1. s_{T_TMC}
2. s_{TI_ABD}
3. s_{T_MCP}
4. s_{T_IP}
5. s_{I_MCP}
6. s_{I_PIP}
7. s_{IM_ABD}
8. s_{M_MCP}
9. s_{M_PIP}
10. s_{R_MCP}
11. s_{R_PIP}
12. s_{MR_ABD}
13. s_{P_MCP}
14. s_{P_PIP}
15. s_{RP_ABD}
16. s_{PALM_ARCH}
17. s_{W_FLEX}
18. s_{W_ABD}

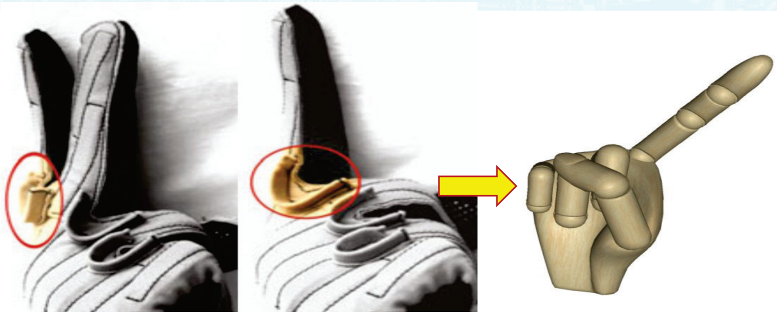


GLOVE CALIBRATION: MULTIPLE SENSORS PER DOF

- Single DOF movements affect multiple sensors
  - Model of one-sensor to one-joint angle isn't accurate
  - Abduction sensors most impacted
- Mapping must take multiple sensors as input
- Need sampling process to obtain calibration data



ADVANTAGES AND CHALLENGES



Advantages

- No occlusion problems
- Large working volume
- May be wireless
- Accurate for flexion

Challenges

- Noise
- Cross coupling
  - Issues for abduction/adduction
- No world space positions
- Some physical encumbrance

STRETCH SENSORS

- How they work
  - Resistive
    - piezoresistive material, an elastic conductive yarn or conductive liquid
  - Capacitive stretch sensors
    - Soft capacitors that can be stretched or squeezed, adjusting the capacitance of the material
  - Commercial and research prototypes
- Advantages:
  - High sensor density (can exceed DOFs)
  - Potential for good accuracy
- Disadvantages:
  - Encumbrance
  - No world-space positioning

[© Stretch Sense]

[©Glauser et al. 2019]

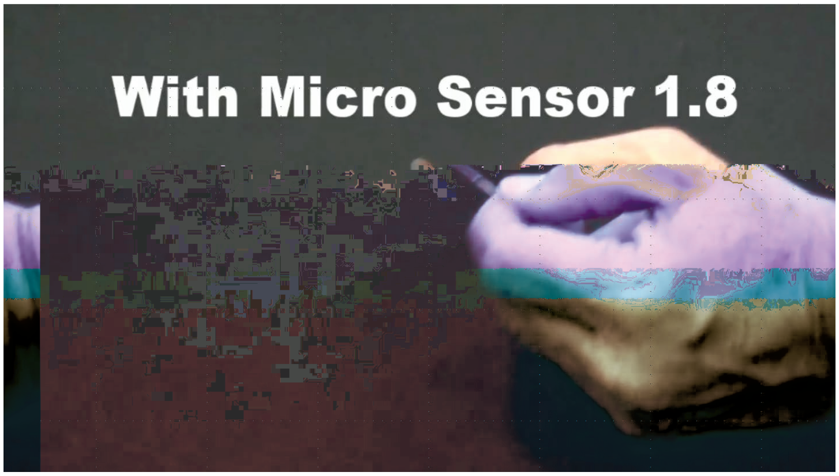
INERTIAL MEASUREMENT UNITS

- How they work:
  - 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer
  - Measure acceleration, rotational rate and orientation
  - Integrate sensor readings to update position/orientation
- Advantages:
  - Potentially very high sample rate
  - Low cost
  - Compact
- Disadvantages:
  - Drift
  - Relatively blocky

[©Noitom Ltd. 2020]

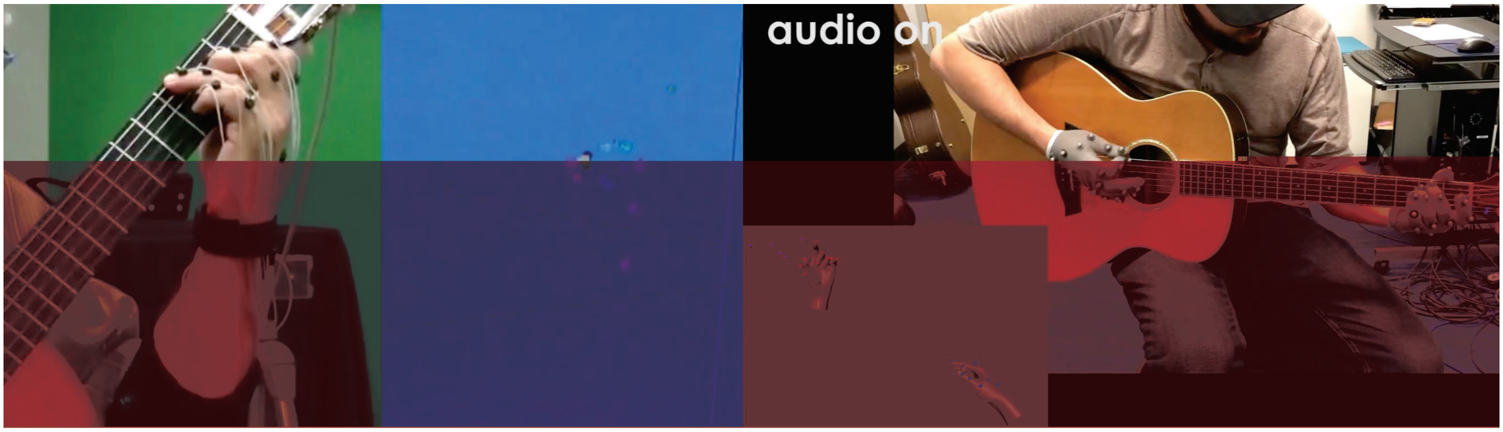
ELECTROMAGNETIC SYSTEMS

- How they work:
  - Low-frequency magnetic field is generated by a transmitter
  - Sensors consist of three orthogonal coils used to measure relative magnetic flux
  - Provide position and orientation, relative to transmitter
- Advantages:
  - Six DOF tracking
  - Sensors have been miniaturized
  - No labeling or occlusion problems
- Disadvantages:
  - Magnetic interference is ubiquitous
  - Calibration



[© Polhemus]

ELECTROMAGNETIC MARKERS



[© Polhemus]

HYBRID APPROACHES

- Combine multiple sensing technologies in a single solution

HYBRID SOLUTIONS: BEND SENSORS WITH IMU

- Combine bend senors with IMUs on each finger

[The Manus Prime II glove for motion capture and VR © Manus]

HYBRID SOLUTIONS: IMU’S AND MAGNETIC

- Combine IMUs with magnetic tracking

[© Rokoko Electronics Aps]

CALIBRATION

- Must map between glove sensor output and finger joint angles
- Calibration defines this mapping
- May prioritize hand shape (FK) or touch accuracy (IK)
- Normally done per subject
  - Variation in hand size and sensor location
- Calibration is increasing in algorithmic sophistication and data requirements
  - Traditional: rely on the linearity of the sensor (calibrate with two data samples)
  - More sophisticated mapping approaches, with increased data needs (e.g. Wang and Neff 2013, tens of samples)
  - Train neural networks with large data needs (e.g. Glauser et al. 2019, 1 million samples)

HAPTIC FEEDBACK

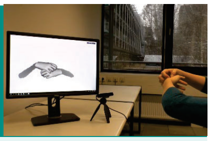
TECHNOLOGIES FOR FINGER MOTION CAPTURE

Markers

- Accurate world space positional tracking
- High framerate
- Relatively easy to use with mature solutions
- Require a dedicated instrumented space
- Labeling and occlusion
- Cost $$$

Gloves

Images



TECHNOLOGIES FOR FINGER MOTION CAPTURE

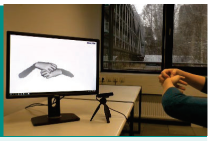
Markers

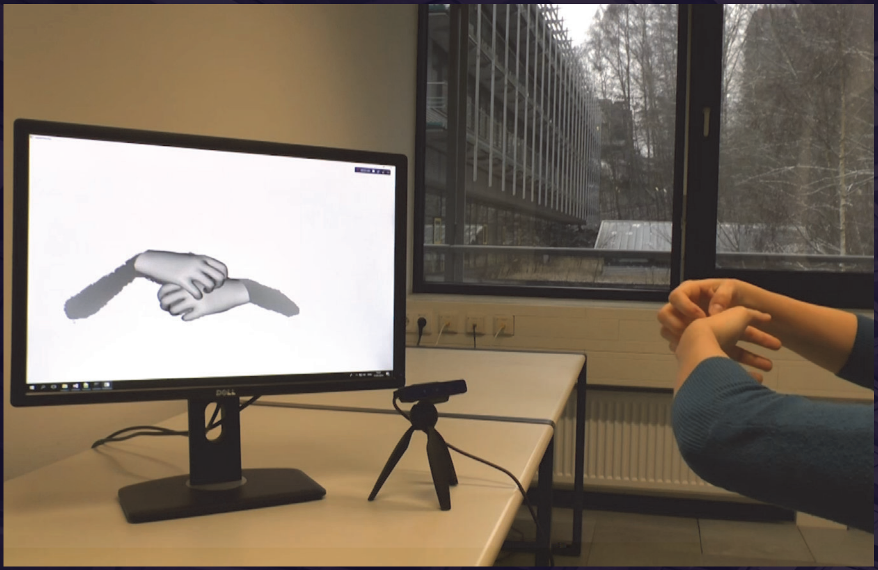
- Accurate world space positional tracking
- High framerate
- Relatively easy to use with mature solutions
- Require a dedicated instrumented space
- Labeling and occlusion
- Cost $$$

Gloves

- Occlusions are not a problem
- Wide range of options
- No capture space requirements
- Flexible working volume
- Users must wear a device
- Calibration may be an issue
- Accuracy varies
- No global position, in general
- Cost $$

Images

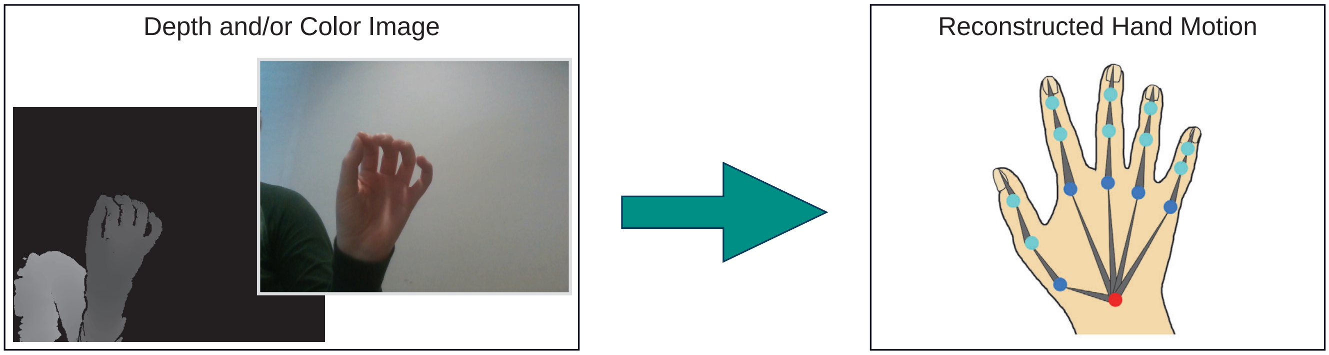




MOTION CAPTURING FINGERS: IMAGE- AND DEPTH-SENSOR-BASED METHODS

Franziska Mueller

GOAL



Although earlier method used calibrated multi-view setups, we will focus on single camera methods:

- easier setup
- more flexible for mobile applications

TECHNOLOGIES FOR FINGER MOTION CAPTURE

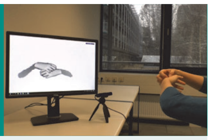
Markers

- Accurate world space positional tracking
- High framerate
- Relatively easy to use with mature solutions
- Require a dedicated instrumented space
- Labeling and occlusion
- Cost $$$

Gloves

- Occlusions are not a problem
- Wide range of options
- No capture space requirements
- Flexible working volume
- Users must wear a device
- Calibration may be an issue
- Accuracy varies
- No global position, in general
- Cost $$

Images



- No instrumentation of the hand
- Global position available (up-to-scale for monocular RGB)
- Easy to setup and use by non-experts, commodity equipment
- Cost $
- Occlusions
- Accuracy lower than marker-based solutions

CLASSES OF METHODS

Generative

- assume availability of a parametric **hand model**
- hand pose is obtained by **minimizing the discrepancy** between the model and the input observation
- does **not have a training stage** -> independent of available data

Discriminative

- assume availability of **large data corpora**
- hand pose is obtained by either database lookup or evaluating a trained **machine learning model**
- can exploit prior **knowledge from data** -> often more robust

Hybrid

- **combination** of generative and discriminative concepts
- try to get **„the best of both worlds“**

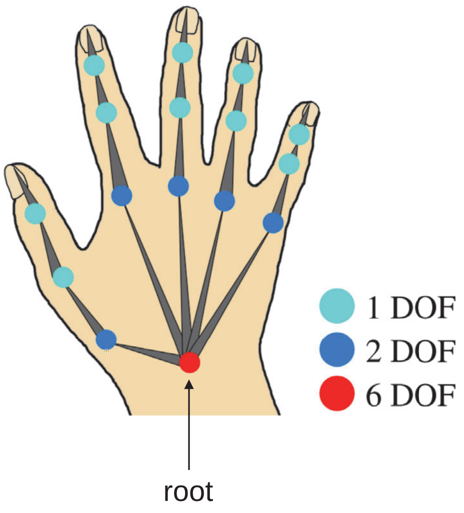
CLASSES OF METHODS

Generative

- assume availability of a parametric **hand model**
- hand pose is obtained by **minimizing the discrepancy** between the model and the input observation
- does **not have a training stage** -> independent of available data

MODELING MOTION: KINEMATIC SKELETONS

- describe hand pose, i.e., configuration of hand bones
- hierarchy (tree) of rigid transforms
- transforms are centered at the joints and are local
  - transform a 3D point from child’s coordinate system to parent’s coordinate system
  - translation = bone length
  - rotation = relative bone rotation
- rotation matrices are constrained by degrees of freedom (DOF) of each joint



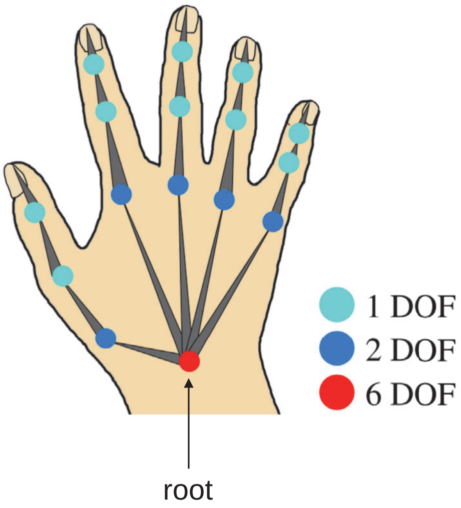
FORWARD KINEMATICS

given values θ for the DOF (pose parameters), calculate the global position of all joints and bones:

- construct the local-to-global transform by walking up the tree

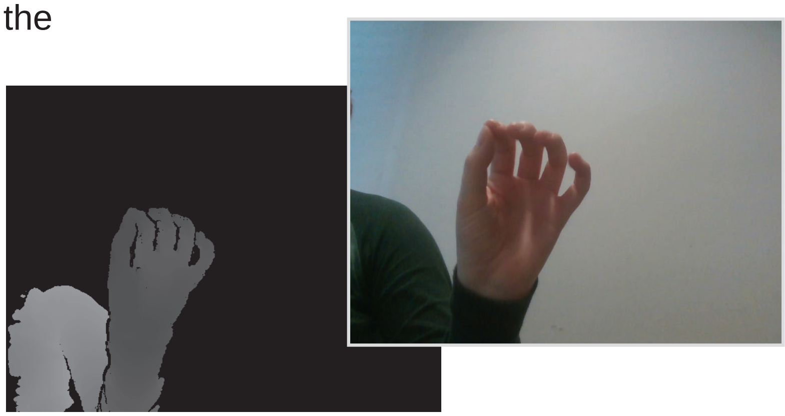
$$T_i^g = \left(\prod_{j \in \mathrm{anc}(i)} T_j^l \right) \cdot T_i^l$$

- multiply it with local coordinates of joints / fingertips to get global position



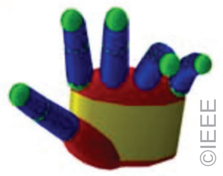
NOW WE HAVE BONES…

- … but they are not directly observable in the input images
- need to model the surface or volume of the hand to measure discrepancy to input



MODELING SURFACE OR VOLUME

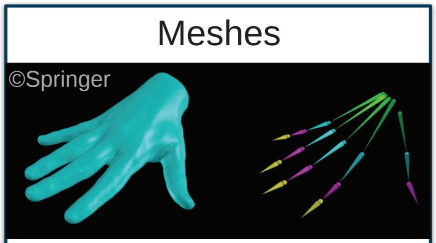
Geometric Primitives



©IEEE

Oikonomidis et al. 2012

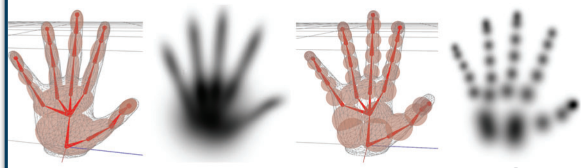
Meshes



©Springer

Tzionas et al. 2016

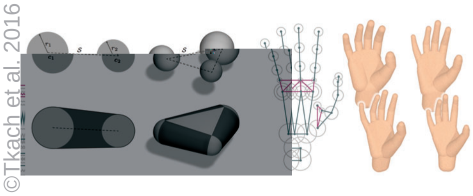
Sum of Gaussians



©IEEE

Sridhar et al. 2014

Sphere Meshes



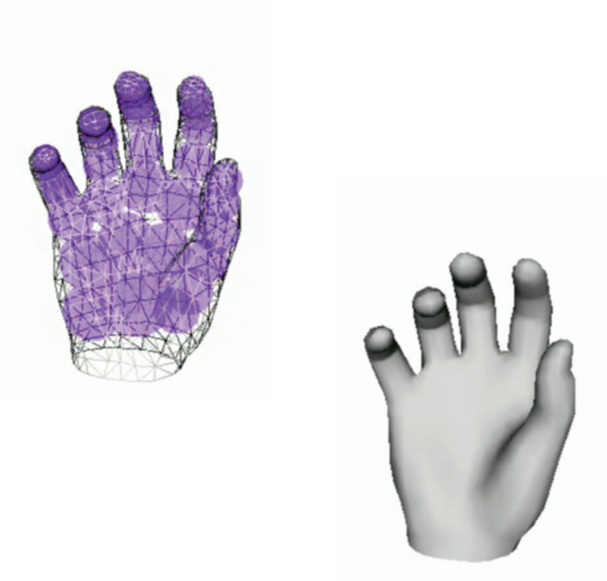
©Tkach et al. 2016

Tkach et al. 2016

Subdivision Surfaces (Taylor et al. 2016),
Articulated Distance Functions (Taylor et al. 2017),
…

COUPLING WITH THE SKELETON

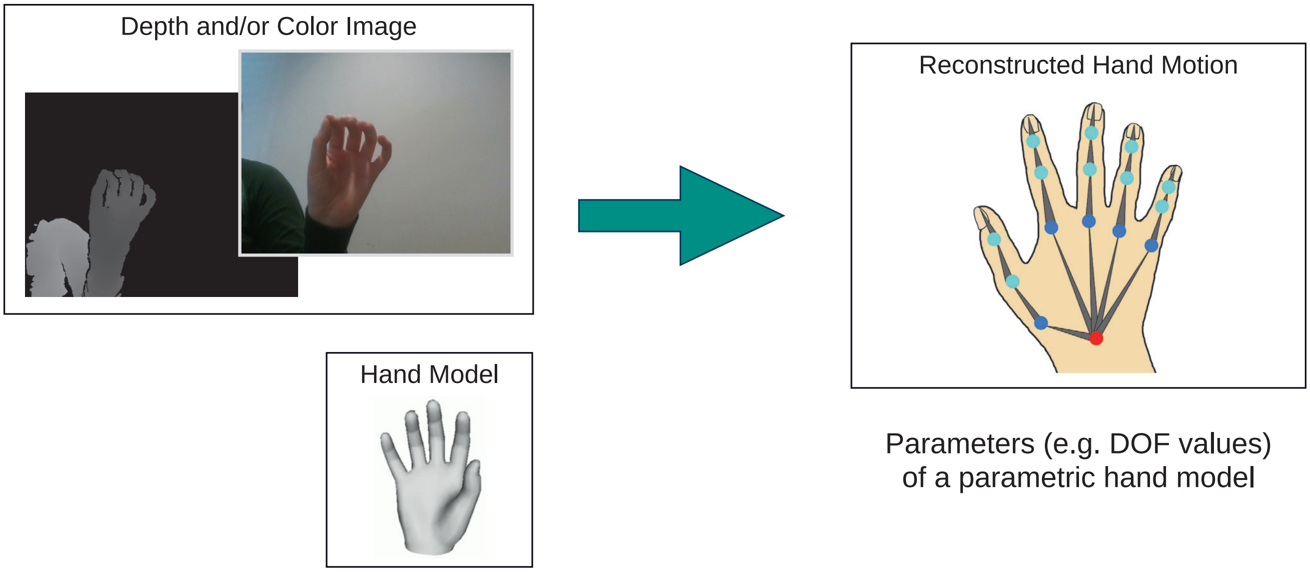
- rigid attachment to a single bone (e.g. for each primitive)
- skinning (e.g. for meshes): vertices can be influenced by multiple bones, especially close to joints where multiple bones meet
- note that positions of vertices or primitives can be easily computed using the local-to-global transforms as described before



COUPLING WITH THE SKELETON

- rigid attachment to a single bone (e.g. for each primitive)
- skinning (e.g. for meshes): vertices can be influenced by multiple bones, especially close to joints where multiple bones meet
- note that positions of vertices or primitives can be easily computed using the local-to-global transforms as described before

GOAL



Depth and/or Color Image

Hand Model

Reconstructed Hand Motion

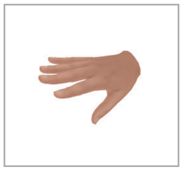
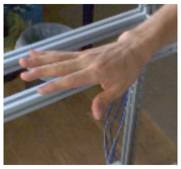
Parameters (e.g. DOF values)
of a parametric hand model

FINDING THE BEST DOF VALUES

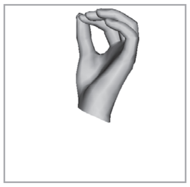
- follows the analysis-by-synthesis principle:
  - (1) we synthesize an initial „image“ using the hand model at some pose hypothesis
  - (2) we calculate a discrepancy measure between the hypothesized image and the input image
  - (3) we try to refine the pose s.t. the synthesized „image“ has smaller discrepancy
  - (4) the final pose is the one that minimizes the discrepancy measure

COMPARING MODEL HYPOTHESIS AND OBSERVATION

- discrepancy measure: usually called objective function or energy
- function of the pose parameters θ

$E(\theta) = ||$  $(\theta) \quad -$  $||$

➔ compare in 2D, usable for depth and color

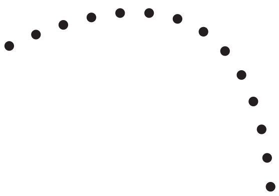
$E(\theta) = ||$  $(\theta) \quad -$ $||$

➔ compare in 3D, usable only for depth

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

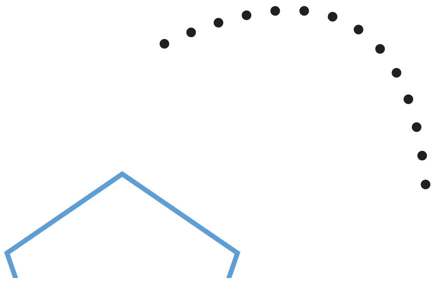


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

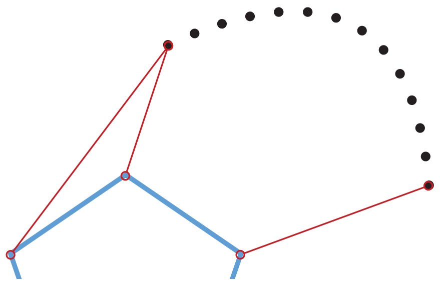


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

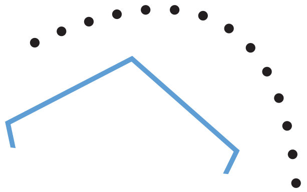


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

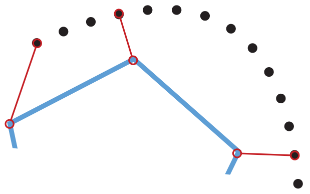


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

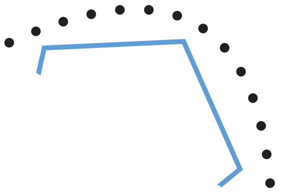


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate

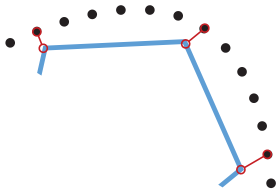


$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

EXAMPLE: FITTING A HAND MESH TO DEPTH

Iterative Closest Point (ICP):

- find closest input point $c(i)$ for each vertex i
- minimize sum of distances to closest point
- iterate



$$E(\theta) = \sum_{i=1}^{|\mathcal{V}(\theta)|} \eta_i \cdot \|\mathcal{V}(\theta)_i - c(i)\|_2^2$$

OPTIMIZATION

- global optimization usually not possible
- differentiability of the energy function enables fast optimization
- derivatives can be used in iterative solvers, e.g., gradient descent, Gauss-Newton, Levenberg-Marquardt,…
- derivatives can be calculated:
  - by hand
  - by optimization libraries, e.g., Ceres
  - your favorite Deep Learning framework
- disadvantage of local solvers: sensitivity to initial pose hypothesis and local optima

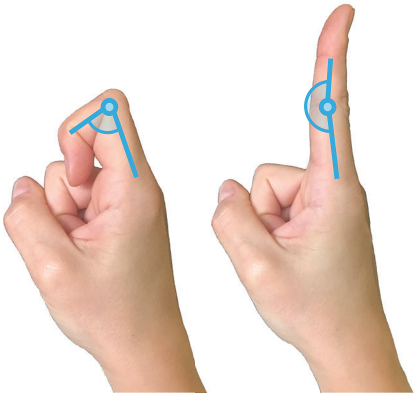
$$\frac{\partial E(\theta)}{\partial \theta_i}$$

THE NEED FOR REGULARIZERS

- objective functions are often non-convex with many local optima
  - it is easy to „get stuck“ at a wrong pose hypothesis
- generative methods do not have a training stage
  - no implicit extraction of prior knowledge from data
- explicit regularizers help to reshape the energy landscape s.t. these wrong poses have higher energy

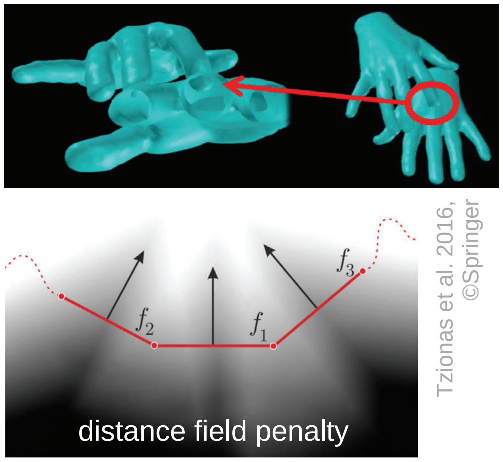
EXAMPLES FOR REGULARIZERS

Joint Angle Limits



e.g. Tkach [2016], Sridhar [2015]

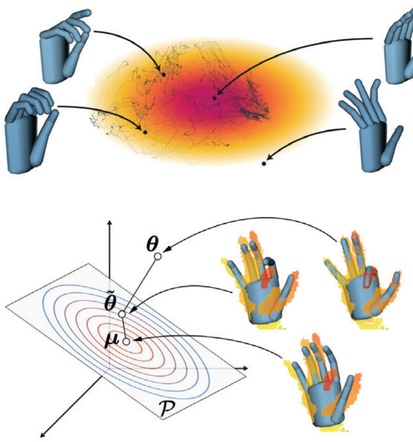
Interpenetration



Tzionas et al. 2016, ©Springer

e.g. Tzionas [2016], Mueller [2019]

Pose Subspaces



Tagliasacchi et al. 2015, ©Wiley

e.g. Tagliasacchi [2015], Tan [2016], Taylor [2016], Mueller [2019]

SUMMARY: GENERATIVE METHODS

- Advantages:
  - independent of data
    - generalization to unseen user / settings
    - no time-consuming / expensive annotations
- Disadvantages:
  - sensitive to initial pose hypothesis
    - can easily „get stuck“ in local minima
  - need for explicit regularizers

CLASSES OF METHODS

Generative

- assume availability of a parametric **hand model**
- hand pose is obtained by **minimizing the discrepancy** between the model and the input observation
- does **not have a training stage** -> independent of available data

Discriminative

- assume availability of **large data corpora**
- hand pose is obtained by either database lookup or evaluating a trained **machine learning model**
- can exploit prior **knowledge from data** -> often more robust

Hybrid

- **combination** of generative and discriminative concepts
- try to get **„the best of both worlds“**

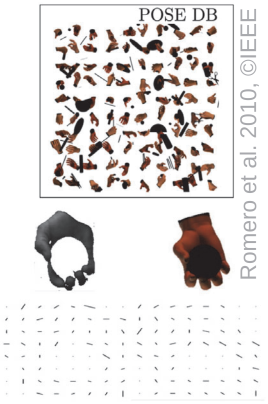
CLASSES OF METHODS

Discriminative

- assume availability of **large data corpora**
- hand pose is obtained by either database lookup or evaluating a trained **machine learning model**
- can exploit prior **knowledge from data** -> often more robust

DISCRIMINATIVE METHODS

Database Search

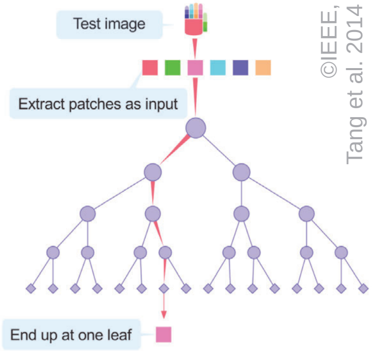


(b) Segmented hand, HOG (c) NN in database, HOG

Romero et al. 2010, ©IEEE

e.g. Romero [2010]

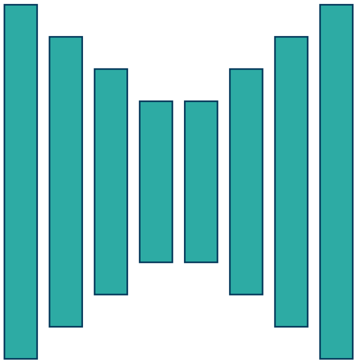
Trained Predictors (pre-deep-learning), e.g. Random Forests



©IEEE, Tang et al. 2014

e.g. Keskin [2012], Tang [2014], Wan [2016]

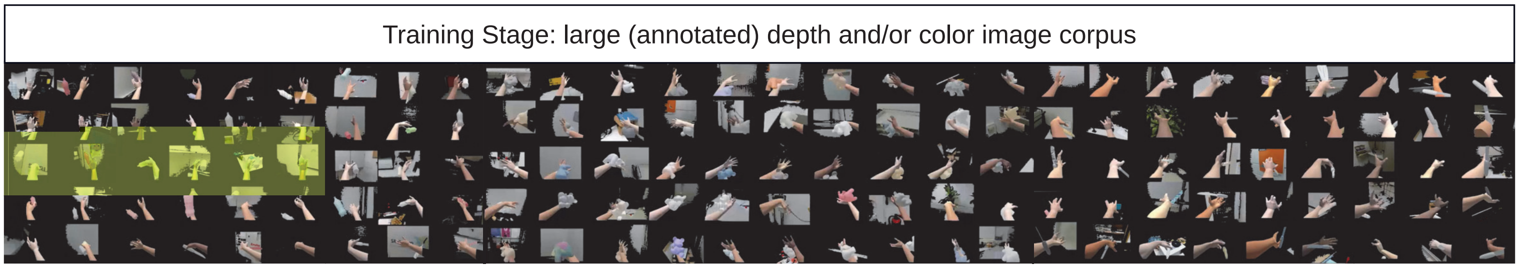
Neural Networks / Deep Learning



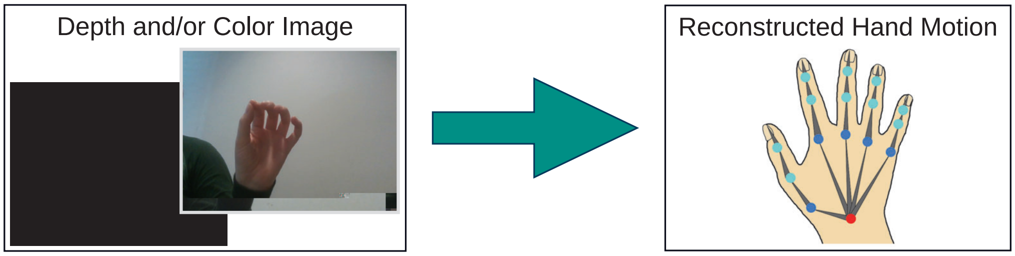
majority of methods in recent years

GOAL

Training Stage: large (annotated) depth and/or color image corpus



Testing Stage:

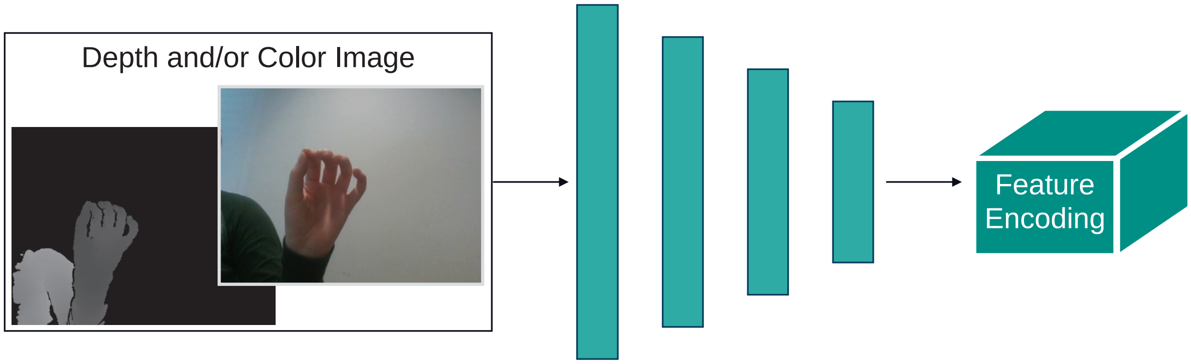


Depth and/or Color Image

Reconstructed Hand Motion

- Joint positions of N_J joints
- Parameters (e.g. DOF values) of a parametric hand model

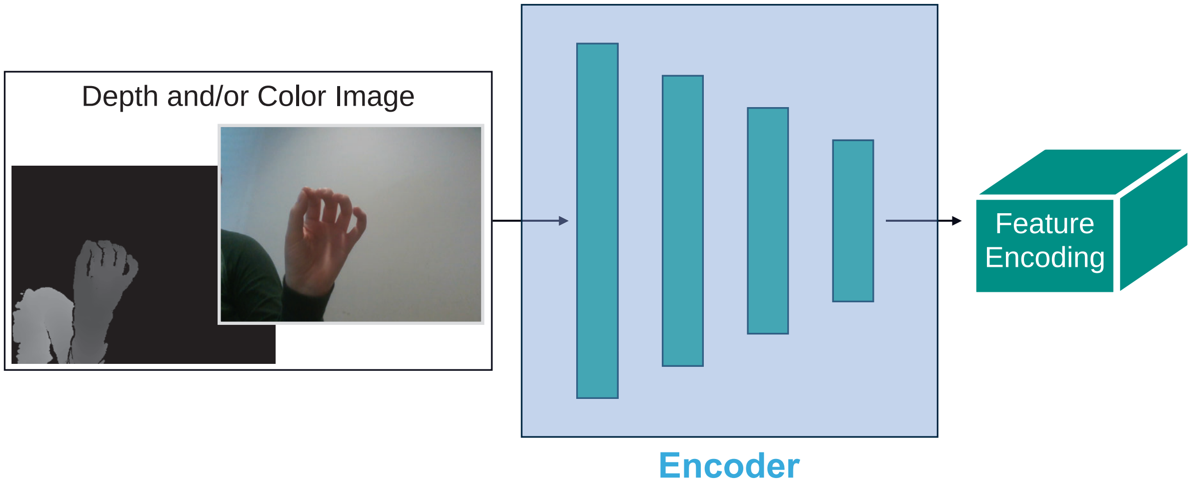
NEURAL NETS: INPUT DATA ENCODERS



Depth and/or Color Image

Feature Encoding

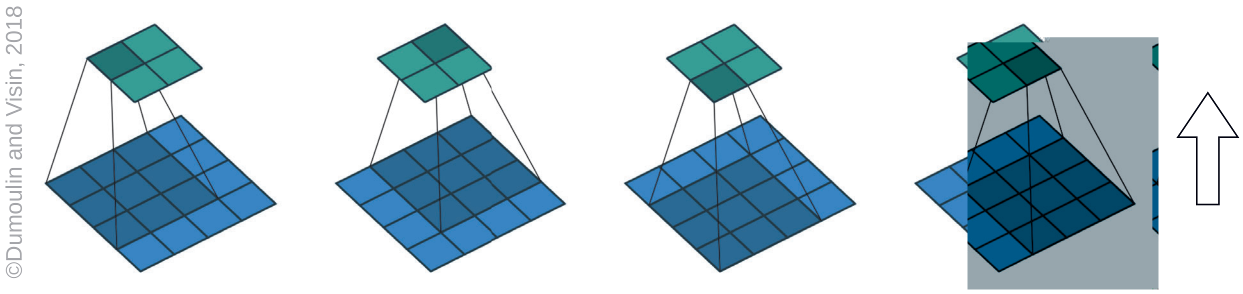
NEURAL NETS: INPUT DATA ENCODERS



NEURAL NETS: INPUT DATA ENCODERS

2D convolutions:

- can be applied to both RGB and depth images



[Dumoulin and Visin, 2018]

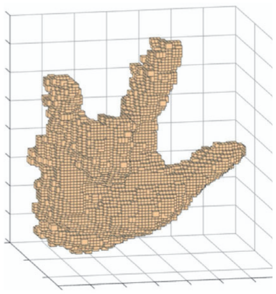
NEURAL NETS: INPUT DATA ENCODERS

[Moon et al. 2018 (hand specific), Milletari et al. 2016 (general)]

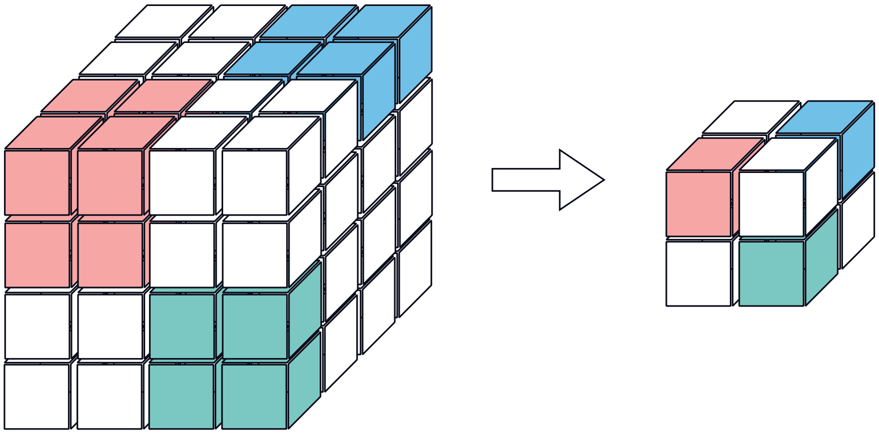
3D convolutions:

- used for voxelized depth data

Moon et al. 2018, ©IEEE



3D voxelized depth map (**V**)

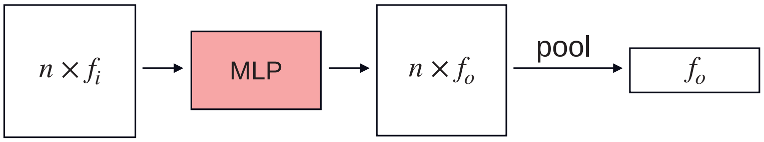


NEURAL NETS: INPUT DATA ENCODERS

[Ge et al. 2018 (hand specific), Qi et al. 2017 (general)]

PointNets (Multi-Layer Perceptron):

- used for point cloud data (e.g. from depth)

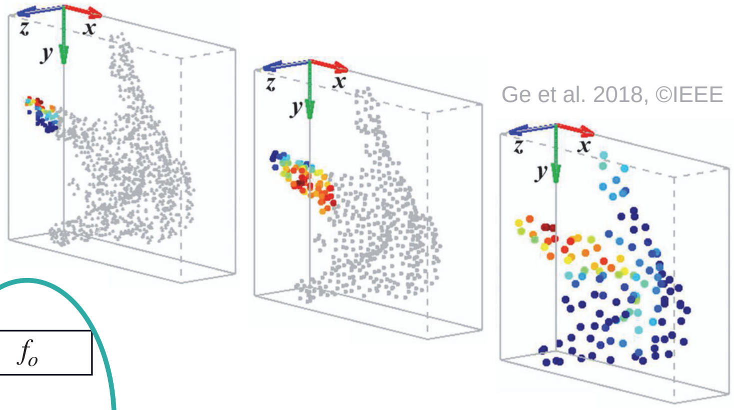


NEURAL NETS: INPUT DATA ENCODERS

[Ge et al. 2018 (hand specific), Qi et al. 2017 (general)]

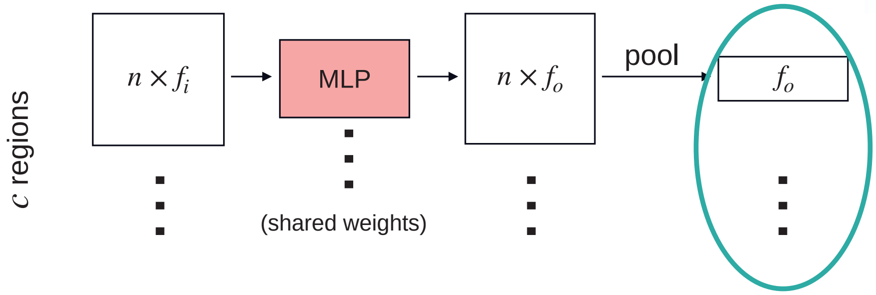
PointNets (Multi-Layer Perceptron):

- used for point cloud data (e.g. from depth)



Ge et al. 2018, ©IEEE

Level l: sample n points around each of c centroids

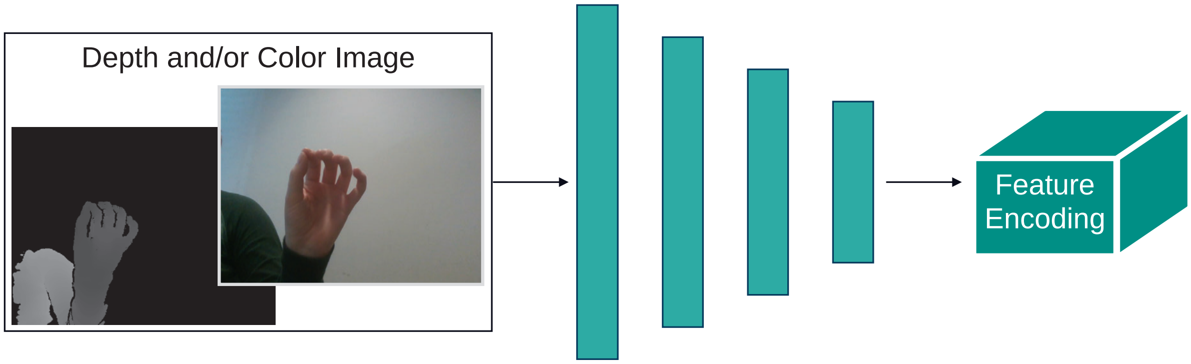


c points of dimensionality f_o are the point cloud for the next level $l+1$

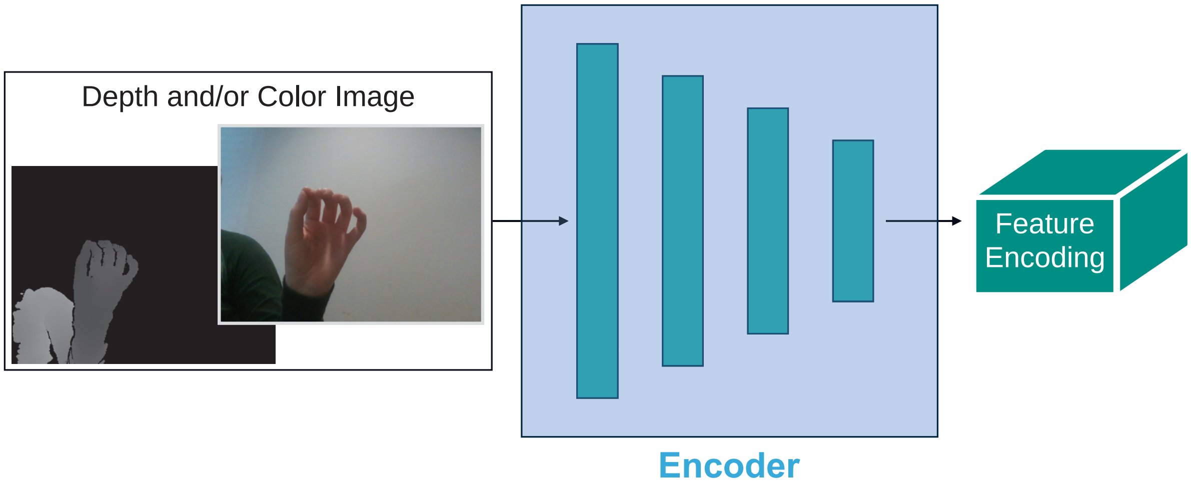
COMPARISON: INPUT DATA ENCODERS

	2D CNN	3D CNN	PointNet
Possible Input Data	RGB and depth images	voxelized depth data	point cloud (e.g. from depth image)
Input Size	$w \times h$	$w \times h \times d$	basic: $n \times 3$ more features: $n \times (3 + f_i)$
Feature Size	$\frac{w}{x} \times \frac{h}{y} \times f$	$\frac{w}{x} \times \frac{h}{y} \times \frac{d}{z} \times f$	$n \times f_o$, possibility to aggregate to a single f_o feature
Core Operation/ Processing Block	2D convolution (slide kernel in 2D)	volumetric convolution (slide kernel in 3D)	Multi-Layer Perceptron

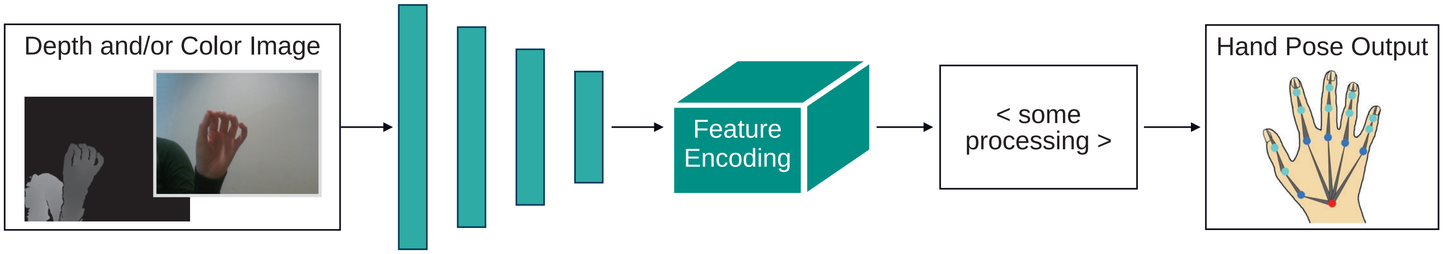
NEURAL NETS: INPUT DATA ENCODERS



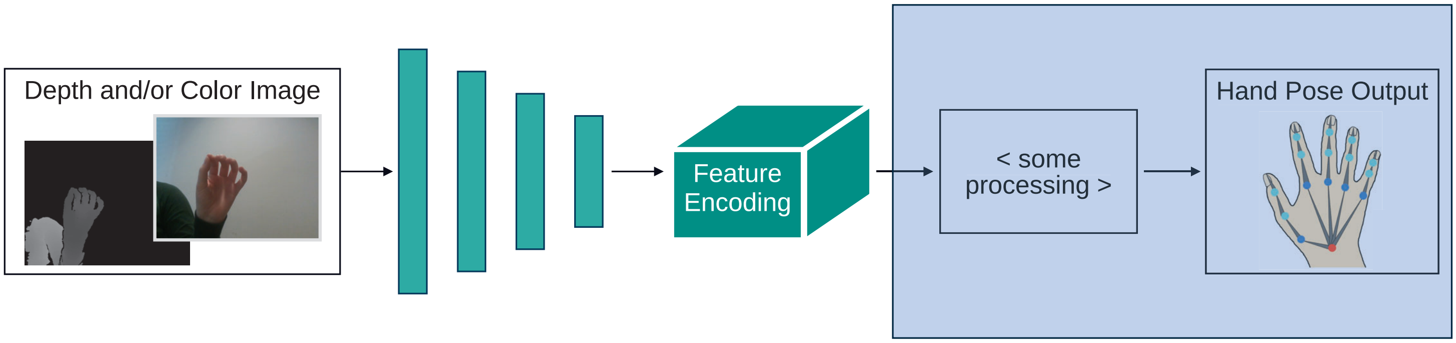
NEURAL NETS: INPUT DATA ENCODERS



NEURAL NETS: OUTPUT REPRESENTATIONS



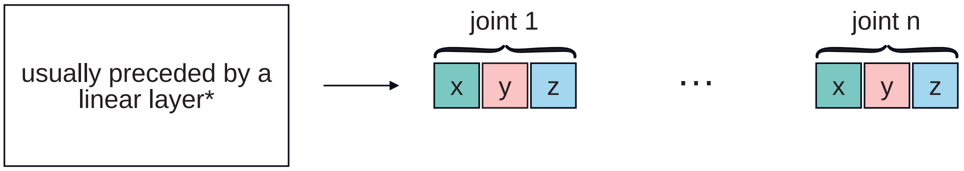
NEURAL NETS: OUTPUT REPRESENTATIONS



NEURAL NETS: OUTPUT REPRESENTATIONS

Vectorized Joint Positions

- output vector of size: $N_J \times 3$
- for monocular RGB, there is depth-scale ambiguity ➔ 3D positions are regressed relative to some reference joint (e.g. wrist or middle MCP)

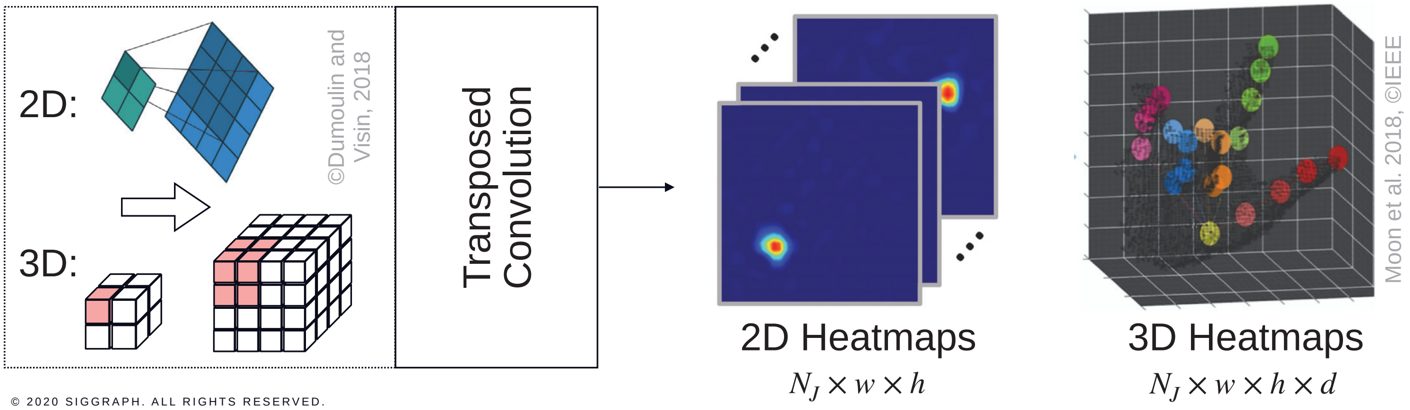


(*it is possible to find convolution kernel sizes and strides to obtain the right dimensionality)

NEURAL NETS: OUTPUT REPRESENTATIONS

Heatmaps (2D / 3D)

- 2D: resolution can be the same as input image or smaller ➔ speed-accuracy trade-off
- 3D: accuracy highly dependent on 3D grid resolution ➔ larger memory cost



©Dumoulin and Visin, 2018

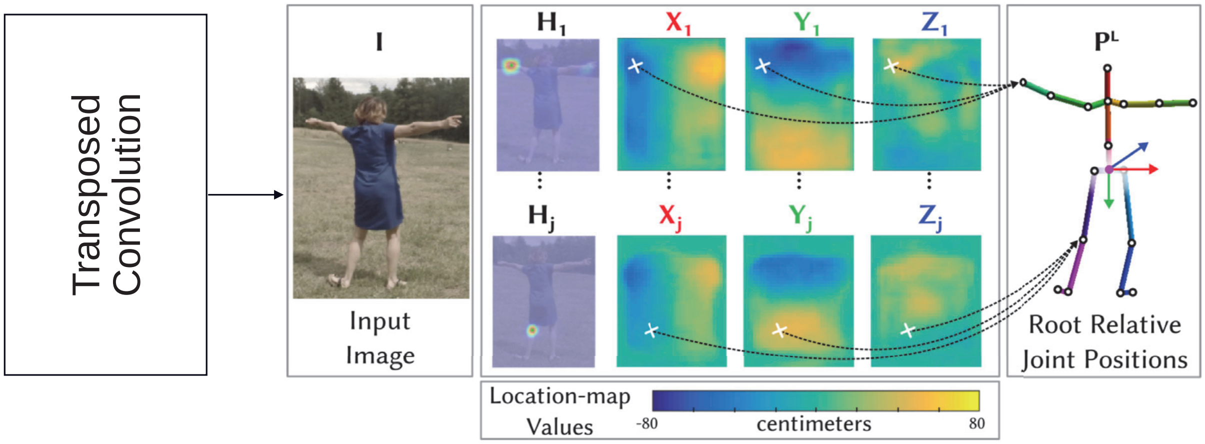
Moon et al. 2018, ©IEEE

NEURAL NETS: OUTPUT REPRESENTATIONS

[Mehta et al. 2017]

Location Maps

- used together with 2D heatmaps to enable 3D pose prediction



Transposed Convolution

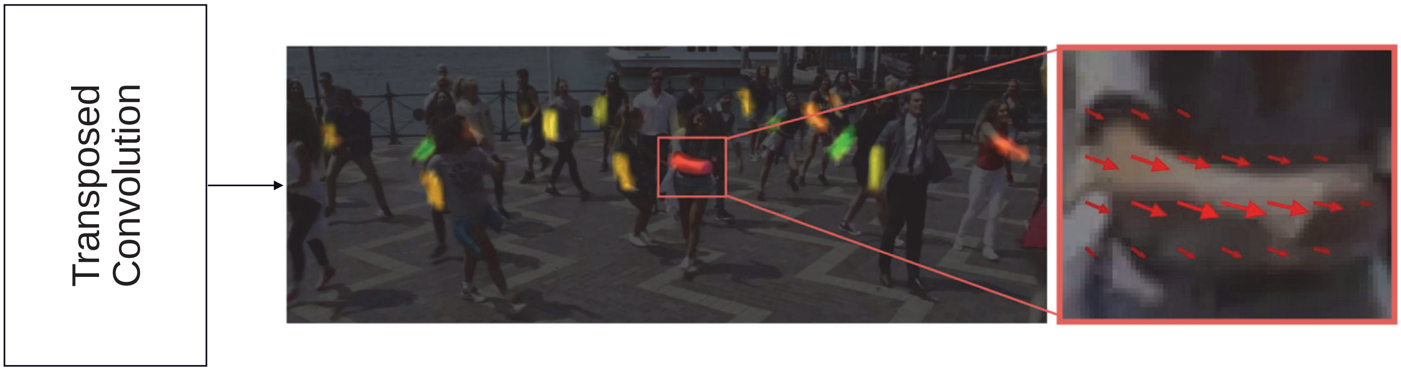
©Mehta et al., 2017

NEURAL NETS: OUTPUT REPRESENTATIONS

[Luo et al. 2018 (3D), Xiang et al. 2019 (3D)]
[Cao et al. 2017 (2D)]

Part Orientation Fields

- encode offsets between child and parent joints



Transposed Convolution

Cao et al. 2017, ©IEEE

COMPARISON: OUTPUT REPRESENTATIONS

	Vectorized Joint Positions	Heatmaps (2D/3D)	Location Maps (LM)	Part Orientation Fields (POF)
Output Size	$N_J \times 3$	2D: $N_J \times w \times h$ 3D: $N_J \times w \times h \times d$	$3N_J \times w \times h$	$3N_J \times w \times h$
Processing	linear layers (+ non-linearities)	transposed convolution (2D/3D)	transposed convolution (2D)	transposed convolution (2D)
Pro	• compact • linear layers encourage pose prior	• closer to input image ➜ better reprojection error	• single read-out per joint (vs. POF)	• takes bones into account
Con	• less spatial correlation to input image	• not compact • less robust to e.g. occlusions • 2D: no 3D pose • 3D: memory cost	• prerequisite: 2D heatmaps • not compact	• prerequisite: 2D heatmaps • not compact • more complicated read-out (vs. LM)

SUMMARY: DISCRIMINATIVE METHODS

- Advantages:
  - implicitly learned prior over training data
  - do not require initialization at test time
- Disadvantages:
  - need for training data
    - less generalization to unseen data
    - time-consuming / expensive annotations

CLASSES OF METHODS

Generative

- assume availability of a parametric **hand model**
- hand pose is obtained by **minimizing the discrepancy** between the model and the input observation
- does **not have a training stage** -> independent of available data

Discriminative

- assume availability of **large data corpora**
- hand pose is obtained by either database lookup or evaluating a trained **machine learning model**
- can exploit prior **knowledge from data** -> often more robust

Hybrid

- **combination** of generative and discriminative concepts
- try to get **„the best of both worlds“**

CLASSES OF METHODS

Hybrid

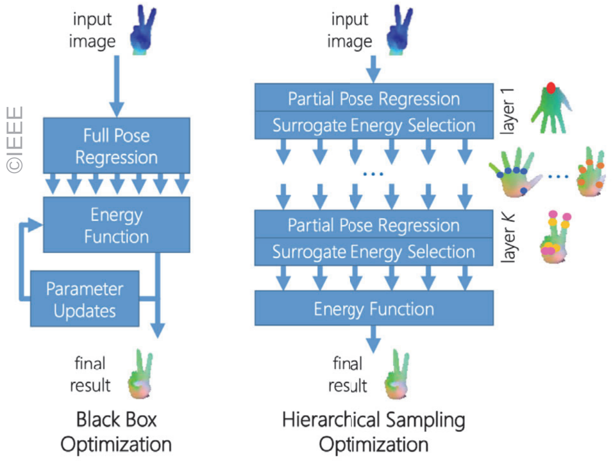
- **combination** of generative and discriminative concepts
- try to get **„the best of both worlds“**

HYBRID METHODS

- designing regressors according to hand model structure
  - easier training & more robustness at test time
- using discriminative output as initialization for generative fitting
  - less sensitive to initial pose
- using discriminative output in an energy term
  - prevent poor local optima
- using a generative hand model inside a neural network, e.g. as non-trainable layer

EXAMPLES: DESIGN & INITIALIZATION

- build regressor hierarchy according to hand model (Tang et al. 2015)



input image

Full Pose Regression

Energy Function

Parameter Updates

final result

Black Box Optimization

input image

Partial Pose Regression

Surrogate Energy Selection

layer 1

...

Partial Pose Regression

Surrogate Energy Selection

layer K

Energy Function

final result

Hierarchical Sampling Optimization

©IEEE

- initialize mesh correspondences instead of closest points (Mueller et al. 2019)

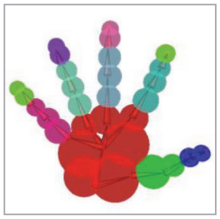
EXAMPLES: USAGE IN ENERGY TERMS

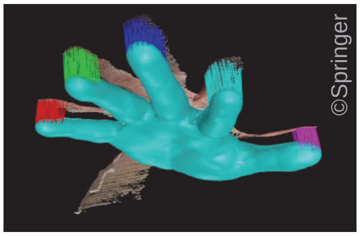
- use regressed part label information for weighting discrepancy in energy function (Sridhar et al. 2015)
- minimize distance between regressed keypoints and model (Tzionas et al. 2016)

Depth Image

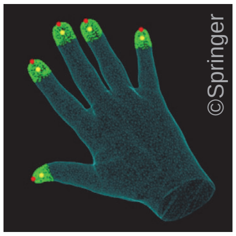
Part Labels

Hand Model





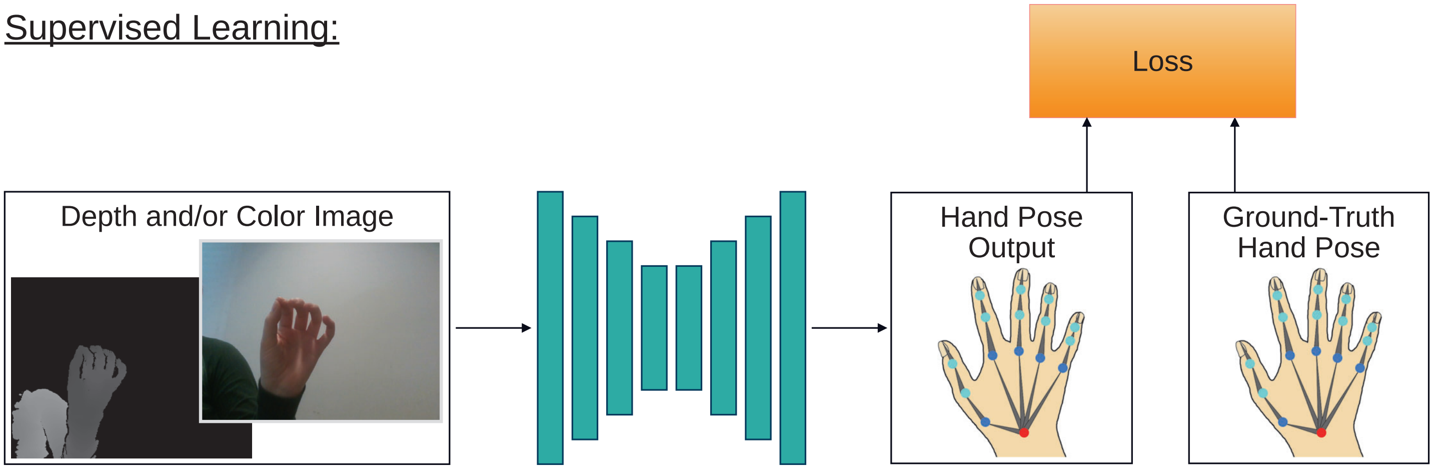
Keypoint Energy Term



Hand Model

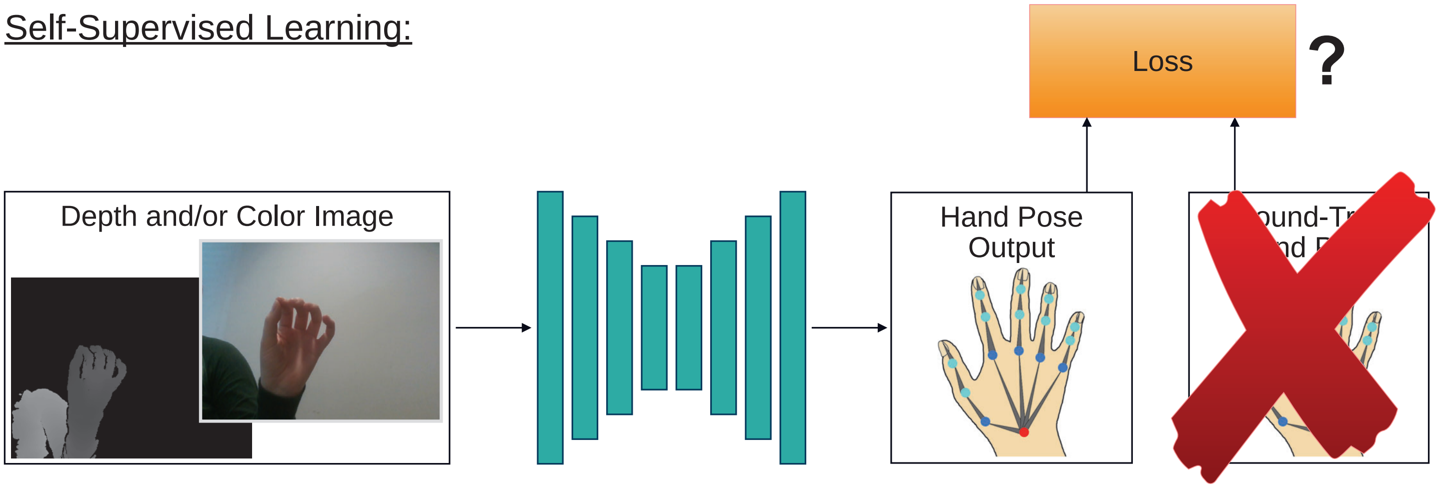
EXAMPLE: SELF-SUPERVISED LOSS

Supervised Learning:



EXAMPLE: SELF-SUPERVISED LOSS

Self-Supervised Learning:



EXAMPLE: SELF-SUPERVISED LOSS

Self-Supervised Learning:

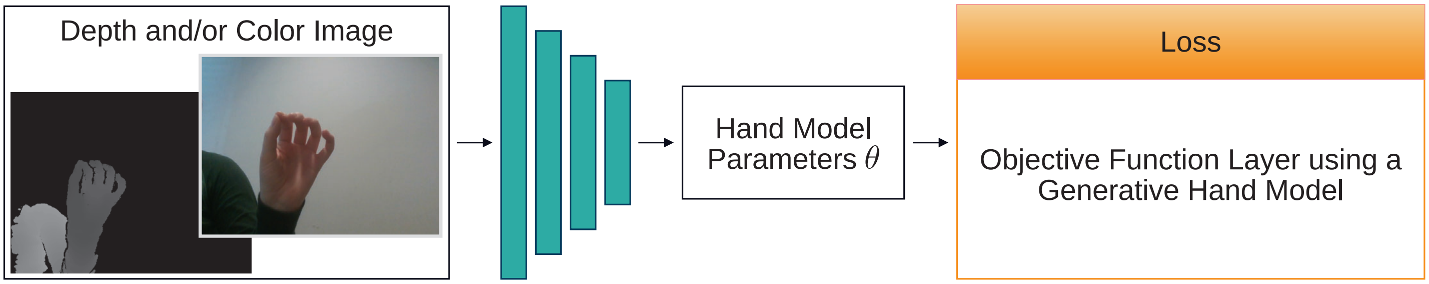
Loss ?

generative model fitting does not need any ground truth

➔ make use of similar objective functions as loss

[e.g. Dibra et al. 2017, Wang et al. 2020]

EXAMPLE: SELF-SUPERVISED LOSS



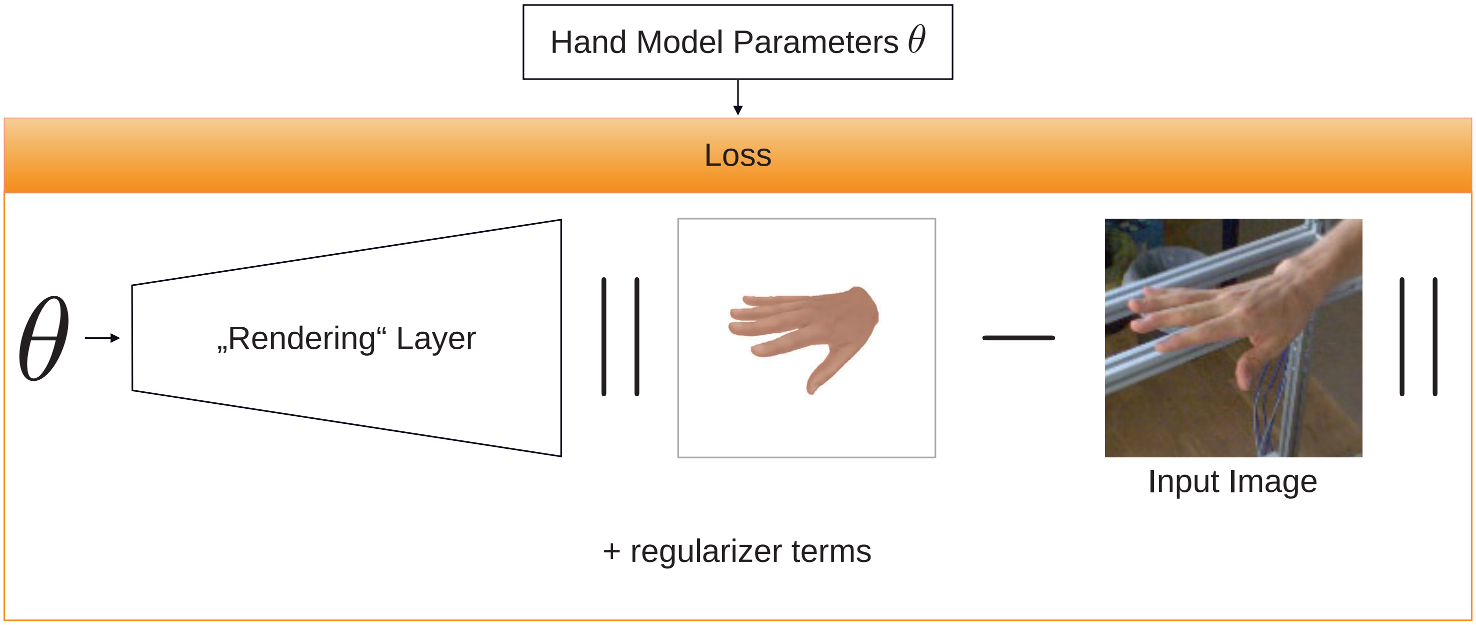
Depth and/or Color Image

Hand Model Parameters θ

Loss

Objective Function Layer using a Generative Hand Model

EXAMPLE: SELF-SUPERVISED LOSS



Hand Model Parameters θ

Loss

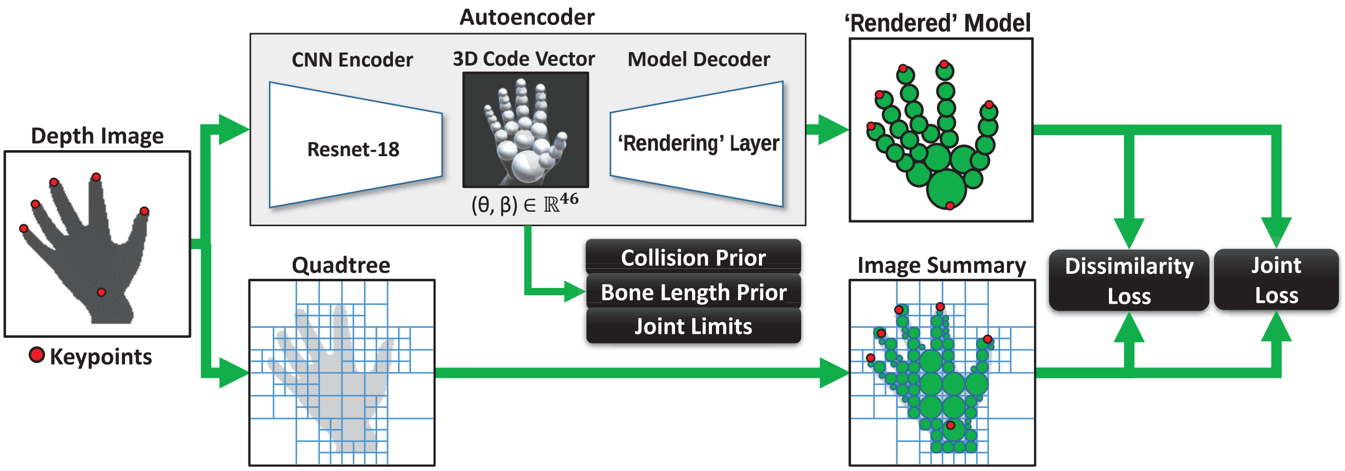
θ „Rendering“ Layer

Input Image

+ regularizer terms

EXAMPLE: SELF-SUPERVISED LOSS

[Wang et al. 2020]



Autoencoder

CNN Encoder

Resnet-18

3D Code Vector

$(\theta, \beta) \in \mathbb{R}^{46}$

Model Decoder

'Rendering' Layer

'Rendered' Model

Depth Image

Keypoints

Quadtree

Collision Prior

Bone Length Prior

Joint Limits

Image Summary

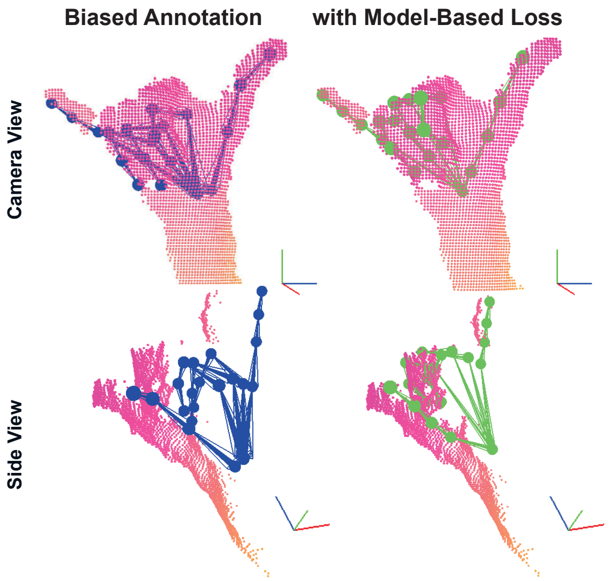
Dissimilarity Loss

Joint Loss

INFLUENCE OF THE SELF-SUPERVISED LOSS

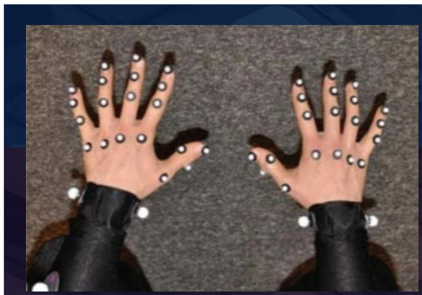
[Wang et al. 2020]

- reduces the amount of annotations that are necessary for training
  - only fingertips (Wang et al. 2020)
  - no keypoints (Dibra et al. 2017)
- the machine learning predictor is learning to explain evidence in the input data
  - counters annotation biases in training datasets



SUMMARY: HYBRID METHODS

- Advantages:
  - often more robust than generative or discriminative methods alone
  - do not solely rely on training data and annotations
  - make knowledge about generative model fitting re-usable
- Disadvantages:
  - might require more complicated implementation



MOTION CAPTURING FINGERS: SUMMARY

Franziska Mueller

TECHNOLOGIES FOR FINGER MOTION CAPTURE

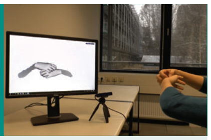
Markers

- Accurate world space positional tracking
- High framerate
- Relatively easy to use with mature solutions
- Require a dedicated instrumented space
- Labeling and occlusion
- Cost $$$

Gloves

- Occlusions are not a problem
- Wide range of options
- No capture space requirements
- Flexible working volume
- Users must wear a device
- Calibration may be an issue
- Accuracy varies
- No global position, in general
- Cost $$

Images



- No instrumentation of the hand
- Global position available (up-to-scale for monocular RGB)
- Easy to setup and use by non-experts, commodity equipment
- Cost $
- Occlusions
- Accuracy lower than marker-based solutions

SIGGRAPH THINK BEYOND
2020 S2020.SIGGRAPH.ORG

HAND MOTION SYNTHESIS

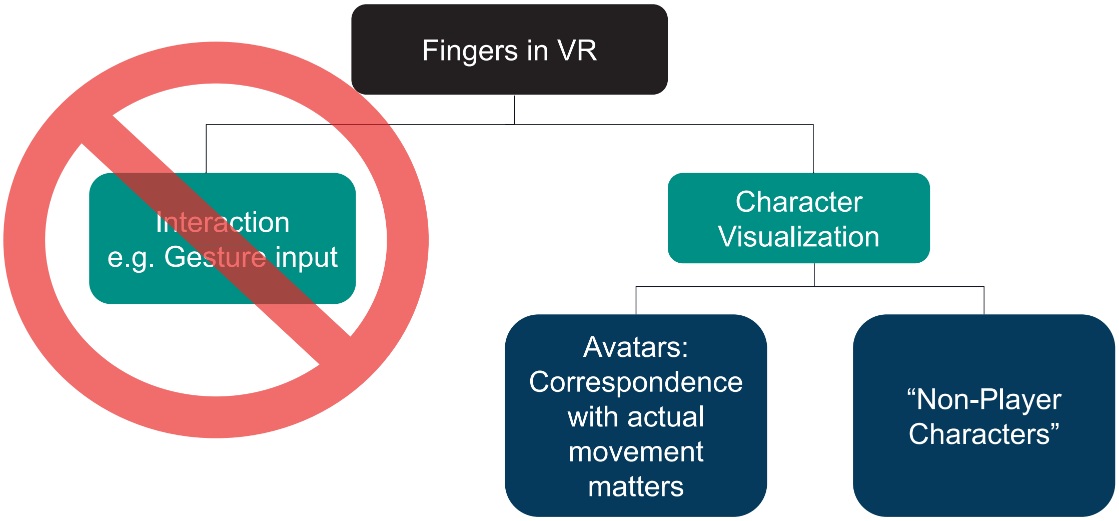
Michael Neff Victor Zordan

SIGGRAPH THINK BEYOND
2020 S2020.SIGGRAPH.ORG

KINEMATIC HAND MOTION SYNTHESIS

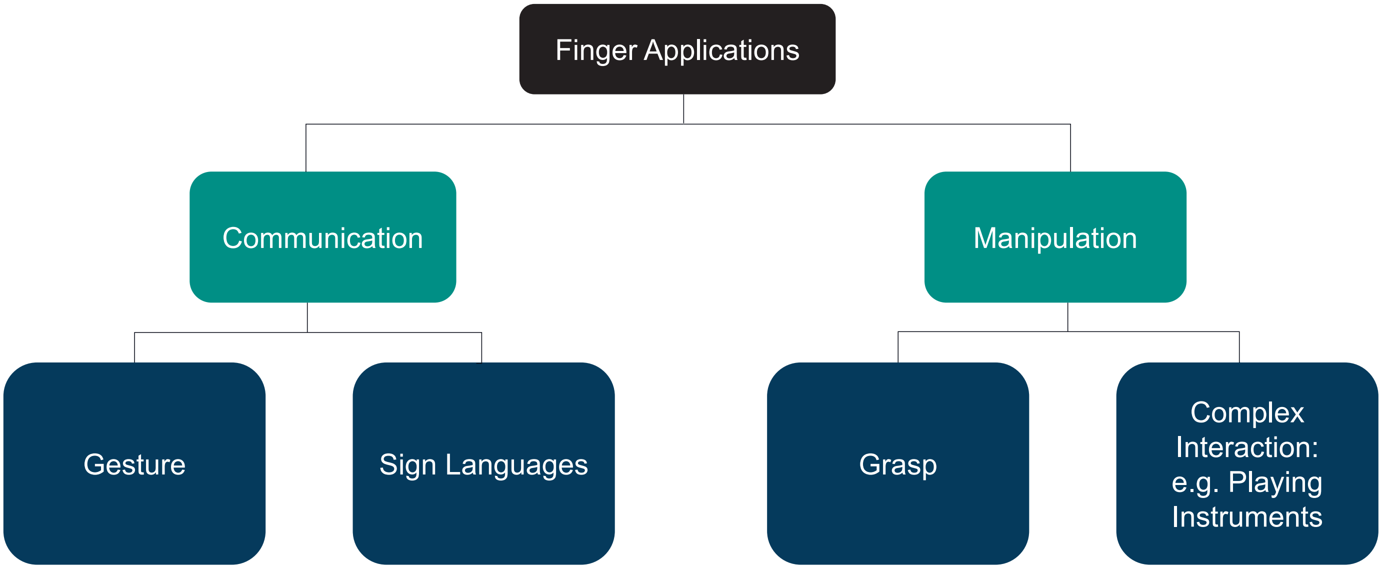
Michael Neff

FINGERS AND VIRTUAL REALITY



- Fingers in VR
  - Interaction e.g. Gesture input
  - Character Visualization
    - Avatars: Correspondence with actual movement matters
    - "Non-Player Characters"

APPLICATION OF FINGER ANIMATION IN VR



- Finger Applications
  - Communication
    - Gesture
    - Sign Languages
  - Manipulation
    - Grasp
    - Complex Interaction: e.g. Playing Instruments

APPROACHES TO HAND ANIMATION

	Kinematic	Data-Driven	Physics-Based
Non-procedural			
Procedural			

APPROACHES TO HAND ANIMATION

	Kinematic	Data-Driven	Physics-Based
Non-procedural			Ragdoll
Procedural			

APPROACHES TO HAND ANIMATION

	Kinematic	Data-Driven	Physics-Based
Non-procedural			Ragdoll
Procedural			Controllers, Constrained Optimization

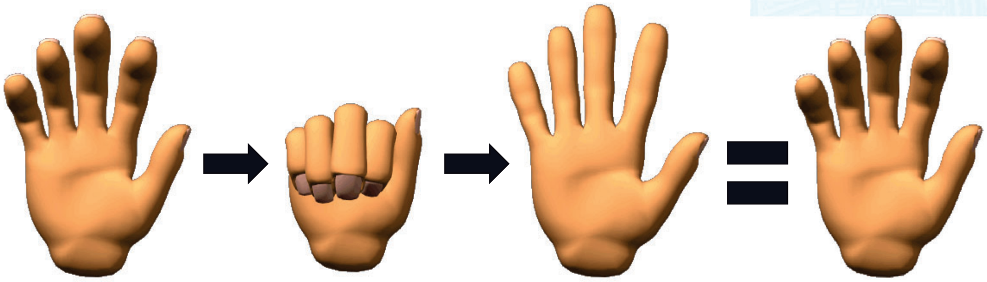
Later…

APPROACHES TO HAND ANIMATION

	Kinematic	Data-Driven	Physics-Based
Non-procedural	Keyframing		Ragdoll
Procedural			Controllers, Constrained Optimization

Later…

KEYFRAMING

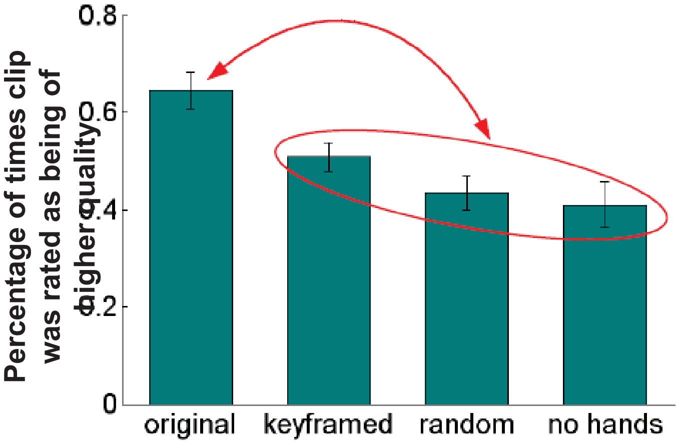


- Can be realistic or heavily stylized
- Depends on skill of animator
- Laborious, but provides full control

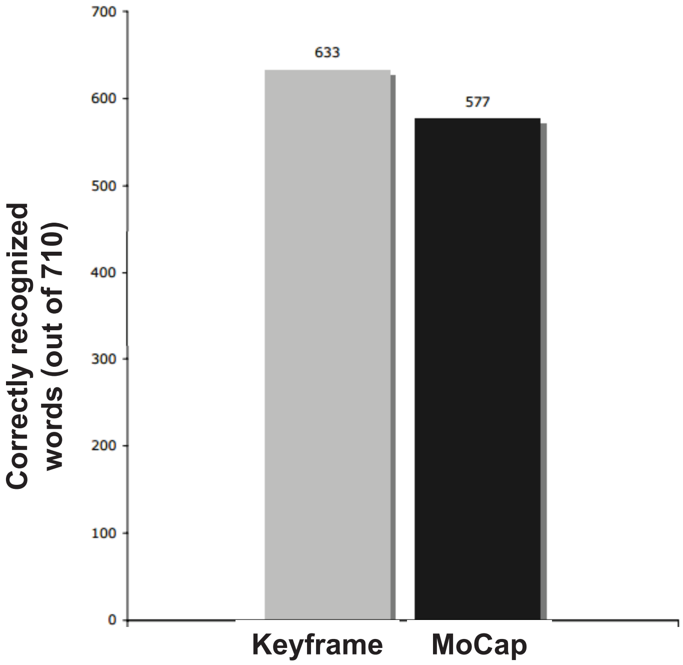
KEYFRAMING

- Device input features a small set of key poses
  - Well suited to keyframing
- Used to augment full body motion that might be generated differently
  - Associate finger pose with gesture phase

- Keyframing quality depends on skill and time
  - Un-restricted keyframes = high quality hand motion
    - Preferred to motion capture [Adamo-Villani 2008]
  - Keyframes restricted to reflect limited time
    - Motion capture preferred [Jörg et al. 2010]



[Jörg et al. 2010]



[Adamo-Villani 2008]

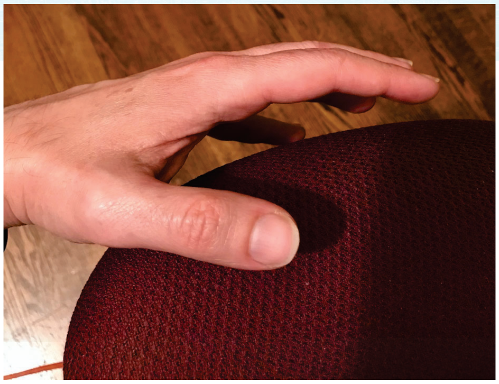
APPROACHES TO HAND ANIMATION

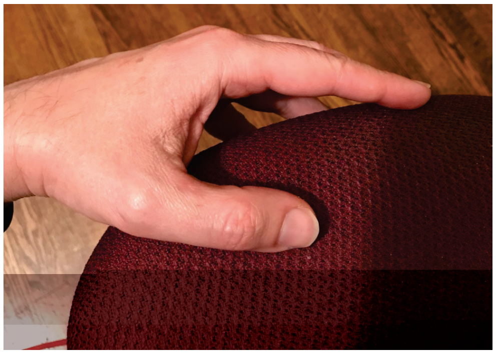
	Kinematic	**Data-Driven**	**Physics-Based**
Non-procedural	Keyframing		Ragdoll
Procedural	Inverse Kinematics		Controllers, Constrained Optimization

Later…

INVERSE KINEMATICS AND HANDS

- Given a set of position and/or orientation constraints, solve for a set of joint angles to achieve those constraints
- Often assume palm location is known, solve for fingers
- Constraints: finger tips, finger bodies, palm, thumb
- Avoid interpenetration
- Solution Methods:
  - Trigonometry
  - Heuristics
  - (Damped) pseudo-inverse of the Jacobian
  - Optimization
  - Data sampling
  - Combinations of the above

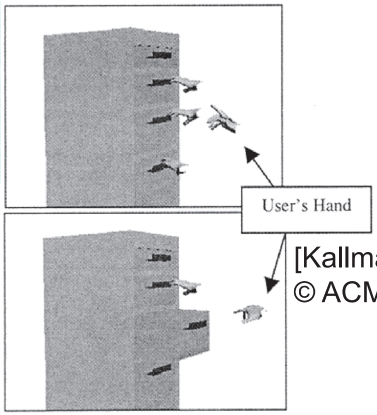




OBJECT AFFORDANCES

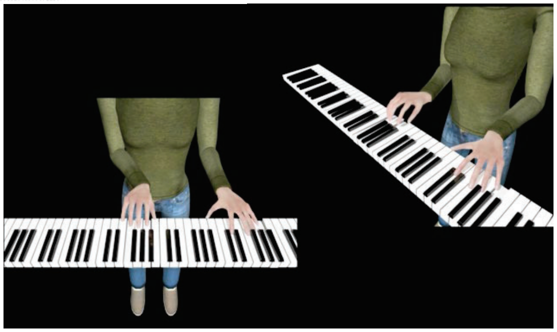
- Object's have affordances:
  - Places where they are touched
  - Orientation constraints
  - Allowable movements
- Information must be encoded
- Constraints must be passed to animation/IK system
- Adaptation must be done on the fly

OBJECT INTERACTION



[Kallmann and Thalmann, 1999 © ACM]

- Must manage hand-object interaction
  - Avoid collision
  - Correctly react to objects
  - “Smart-objects” know their affordances
  - Musical Instruments
- Technologies
  - Hierarchical motor control
  - Inverse kinematics
- Triggered Animations ⟷ Direct Interaction



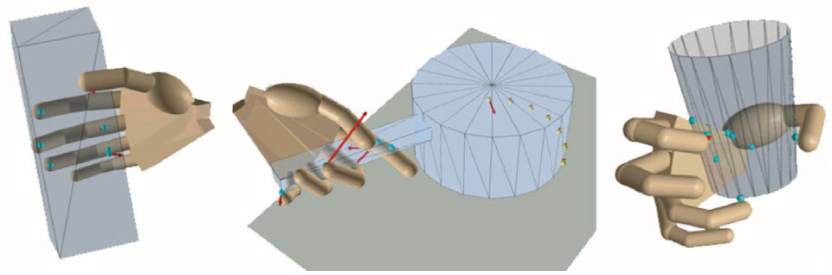
[Zhu et al. 2013]

GRASP AS A GAME MECHANIC

- The Climb
  - Rock climbing VR game
- Lone Echo
  - Grasp to move through space environment
  - Algorithm:
    - Pre-author poses for complicated touch like holding a gun
    - Otherwise, search for objects nearest palm
    - Apply IK to have fingers make contact
    - Allow repositioning of hand, but minimize
    - Discarded physics:
      - Concerns about performance
      - Need for graceful failure

OBJECT INTERACTION WITH PHYSICS

- Examine the physical requirements of supporting objects
- Solve for possible contact points that meet physical requirements
- May assume object motion is known
- Examples [Ye and Liu, 2012], [Mordatch et al., 2012]



(Ye and Liu, 2012)

APPROACHES TO HAND ANIMATION

	Kinematic	Data-Driven	Physics-Based
Non-procedural	Keyframing	Motion Playback	Ragdoll
Procedural	Inverse Kinematics		Controllers, Constrained Optimization

Later…

DATA-DRIVEN, NON-PROCEDURAL

- Record separate hand motion, timed to body motion
- Combine two data streams for final animation
- Pros:
  - High quality motion
- Cons:
  - Labor intensive
  - Inflexible (must record all necessary motion, with appropriate timing to match body motion)

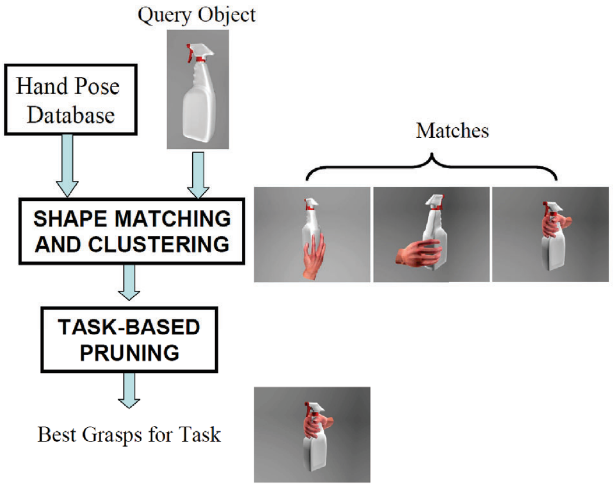
APPROACHES TO HAND ANIMATION

	Kinematic	**Data-Driven**	**Physics-Based**
Non-procedural	Keyframing	Motion Playback	Ragdoll
Procedural	Inverse Kinematics	Statistical Pose Models, Motion Repurposing	Controllers, Constrained Optimization

Later…

DATA-DRIVEN, PROCEDURAL: TOUCH INTERACTION

- Data-Driven Grasp Synthesis (Li et al. 2007)
  - Capture a database of people interacting with objects
  - At runtime, shape match against database to find potential poses
  - Prune based on grasp quality measure
  - Apply best pose



Query Object

Hand Pose Database

SHAPE MATCHING AND CLUSTERING

Matches

TASK-BASED PRUNING

Best Grasps for Task

[Li et al. 2007, © IEEE]

DATA-DRIVEN, PROCEDURAL: TOUCH INTERACTION

- Data-Driven Grasp Synthesis (Li et al. 2007)
  - Capture a database of people interacting with objects
  - At runtime, shape match against database to find potential poses
  - Prune based on grasp quality measure
  - Apply best pose
- Data-Driven hand interaction with remote subjects (Lee et al. 2017)
  - Identify touch locations for important hand poses
  - At run time, identify type of touch
  - Query touch locations
  - Adapt remote participant's pose with IK to maintain proper contact

[© Youjin Lee 2017]

DATA-DRIVEN, PROCEDURAL: DIMENSIONALITY REDUCTION

- High dimensionality of hand pose
  - Fingers have a large number of degrees of freedom (20+)
  - Actual hand poses lie in a lower dimensional space
    - i.e. there are correlations between joints
    - Data is a good way to capture this
- Approaches
  - PCA
  - Non-linear dimensionality reduction

HAND-OVER ANIMATION

- Merge finger animation with pre-existing full body motion
- Reflects challenges of simultaneous finger capture
- May record partial finger motion to guide process

ADAPT HAND MOTIONS TO FIT BODY MOTIONS

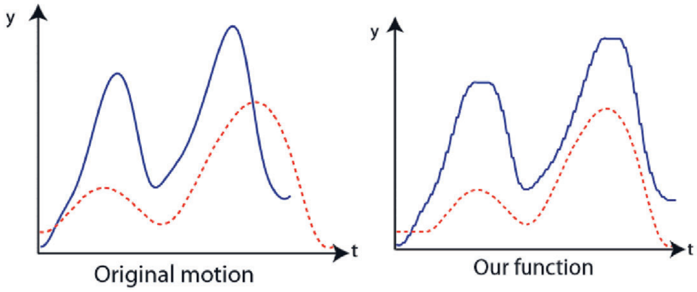
Capture body and finger motions separately so that the hand motions can be captured in a smaller area [Majkowska et al. 2006]

Four markers on hand, wrist, and forearm present in both captures

[©Majkowska et al. 2006]

ADAPT HAND MOTIONS TO FIT BODY MOTIONS

- Motion from hand and body aligned in three steps:
  - align movement phases using dynamic time warping (DTW) and acceleration and velocity profiles
  - align frames within phases with DTW based on the angle between forearm and palm of the hand
  - smooth resulting motions



[©Majkowska et al. 2006]

GENERAL APPROACH

- **Segment motion**
  - poses/frames
  - movement phases
- Choose match in a database
  - nearest neighbor
  - rule based
  - optimization function
- Adapt motion to fit final motion
  - dynamic time warping
  - smoothing
  - physics based simulation

GENERAL APPROACH

- Segment motions
  - poses/frames
  - movement phases
- **Choose match in a database**
  - nearest neighbor
  - rule based
  - optimization function
- Adapt motion to fit final motion
  - dynamic time warping
  - smoothing
  - physics based simulation

GENERAL APPROACH

- Segment motions
  - poses/frames
  - movement phases
- Choose match in a database
  - nearest neighbor
  - rule based
  - optimization function
- **Adapt motion to fit final motion**
  - dynamic time warping
  - smoothing
  - physics based simulation

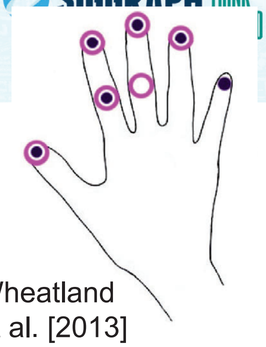
EXAMPLE USES

- Improve motion capturing of data
- Augment body motions with hand motions
- Compute parameters for procedural animation
- Animate conversational characters based on text or speech

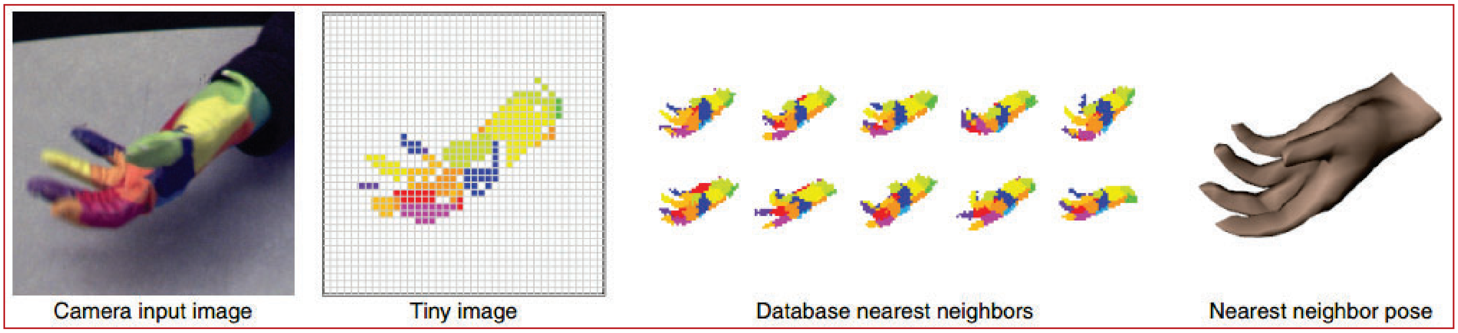
IMPROVE MOTION CAPTURING

- Retrieve high resolution finger motion
  - Based on reduced marker set with optical motion capture [Kang et al. 2012, Wheatland et al. 2013]
  - Based on 3D hand posed with a glove-based input [Wang and Popović 2009]

Kang et al. [2012]



Wheatland et al. [2013]



Camera input image

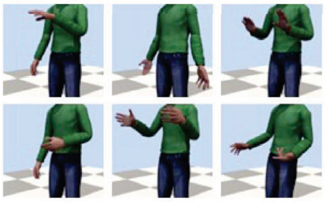
Tiny image

Database nearest neighbors

Nearest neighbor pose

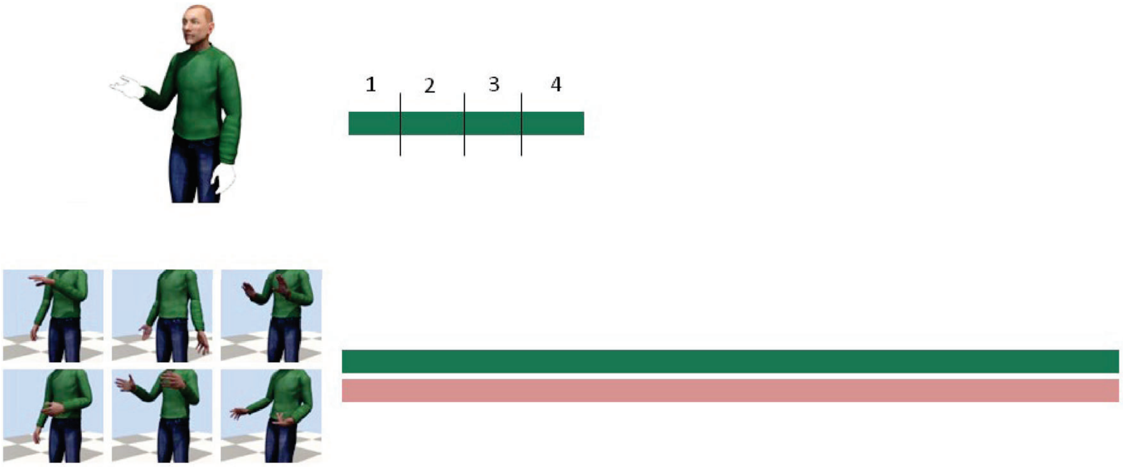
[Wang and Popović 2009, ©ACM]

AUGMENT BODY MOTIONS WITH HAND MOTIONS



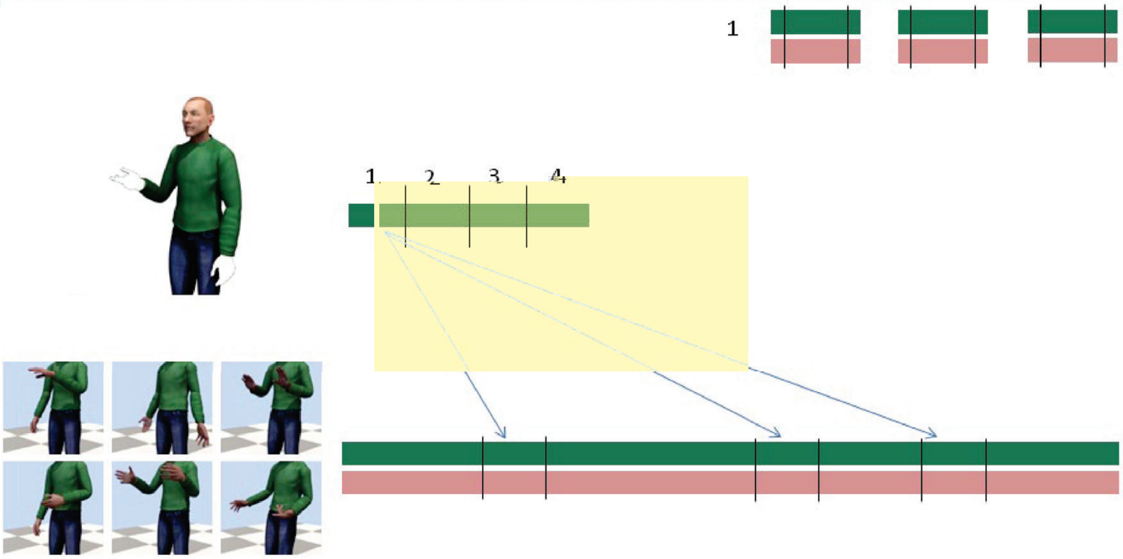
Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS



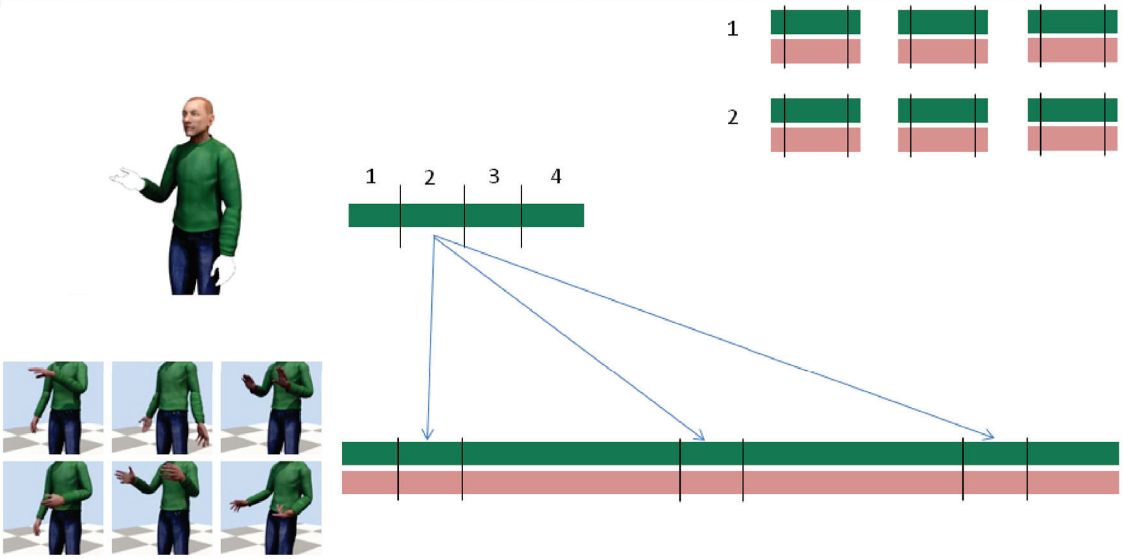
Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS



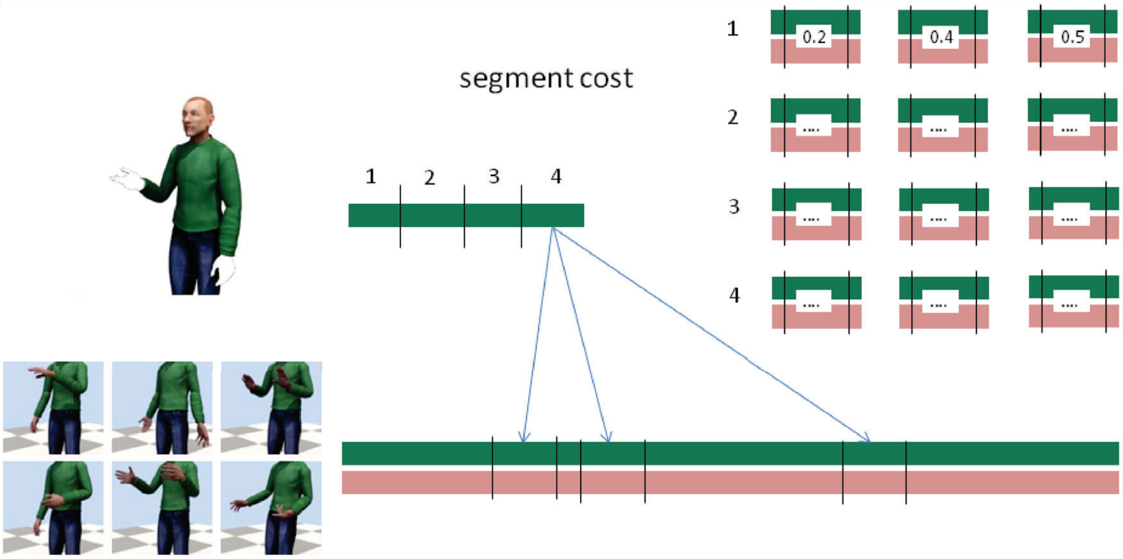
Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS



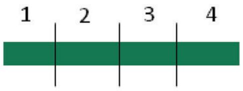
Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS

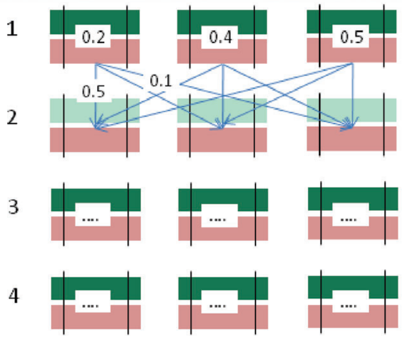

segment cost

Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS

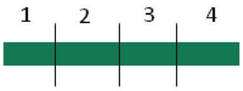


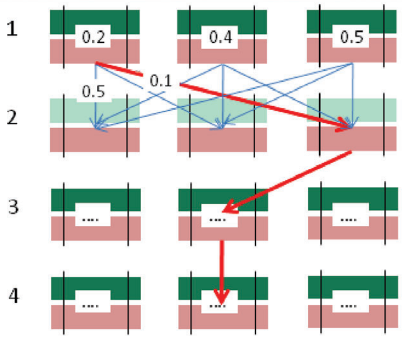
transition cost



Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS

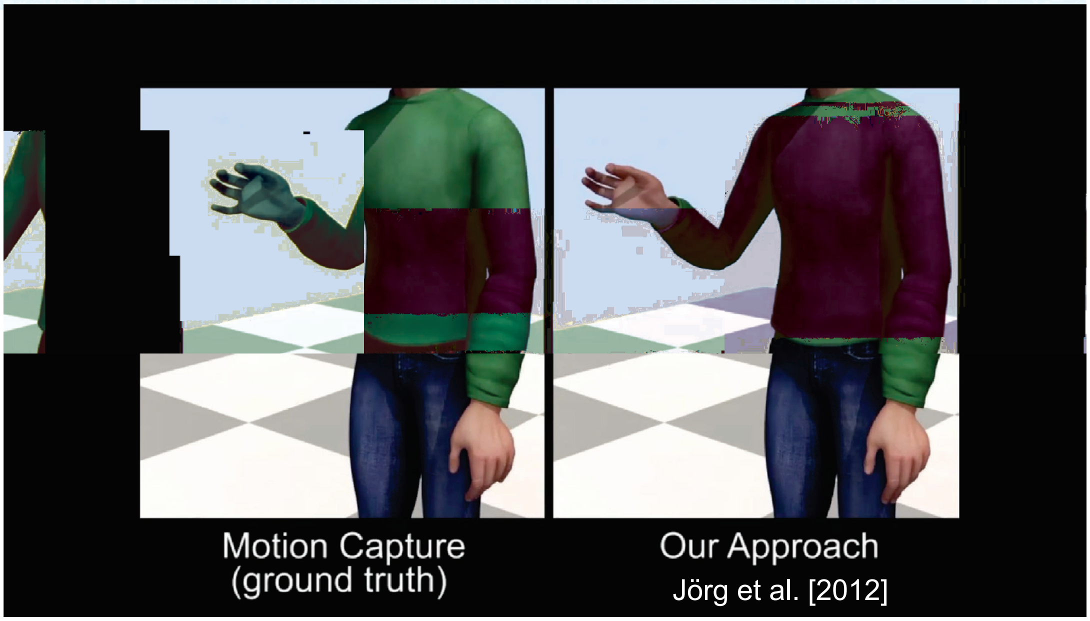




shortest path: Dijkstra's algorithm

Jörg et al. [2012]

AUGMENT BODY MOTIONS WITH HAND MOTIONS



Jörg et al. [2012]

SUMMARY

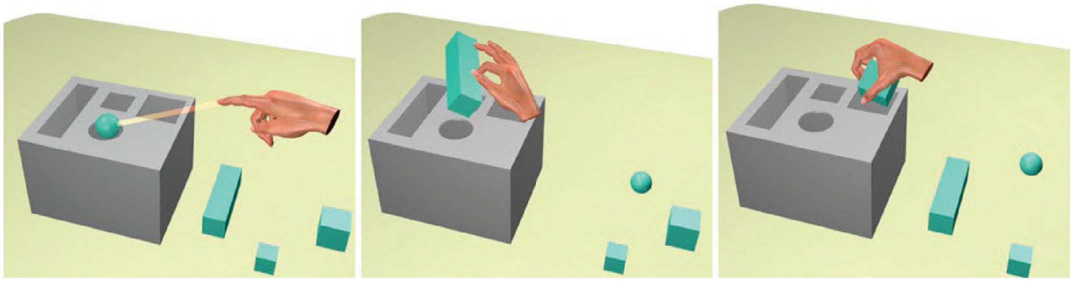
- Keyframing
  - Excellent control, high manual labor cost for quality results
- Procedural, Kinematic
  - Flexible, fast and can be reasonable quality
  - Generally lower quality than data, more manual labor
- Motion capture
  - High motion quality, but requires good data of needed motions
  - Difficult to adapt motion to new situations

PHYSICAL MODELING

Victor Zordan

Why use physics-based hands in VR?

Dextrous, Intuitive, Immersive Manipulation



Raycast “select” Data-driven grasp Physical grasp

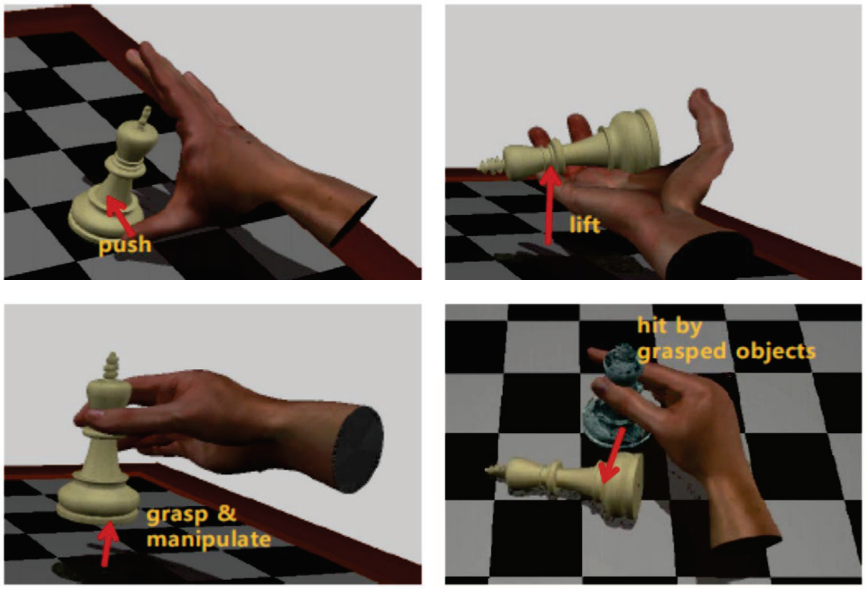
©IEEE [Tian et al. 2019]

Why use physics-based hands in VR?

Motivation

Physics provides a known "language" for interaction

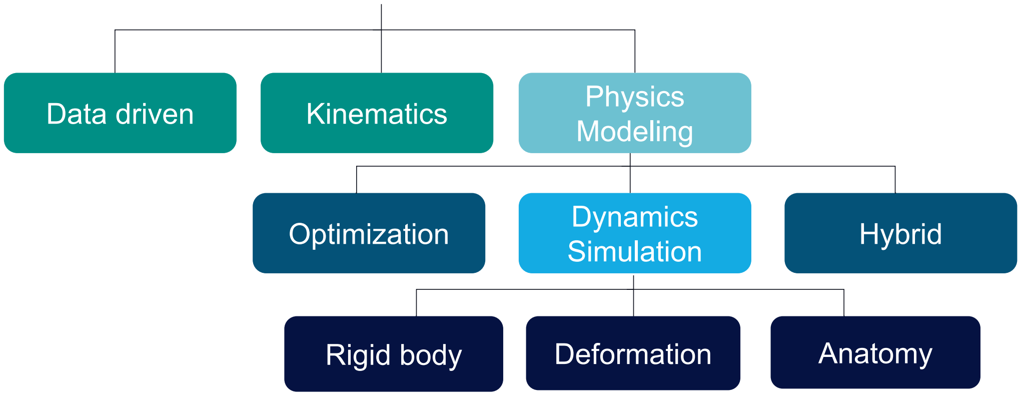
Force as natural interface



©IEEE [Kim and Park 2015]

Introduction to physical modeling for hands

Taxonomy



Data driven

Kinematics

Physics Modeling

Optimization

Dynamics Simulation

Hybrid

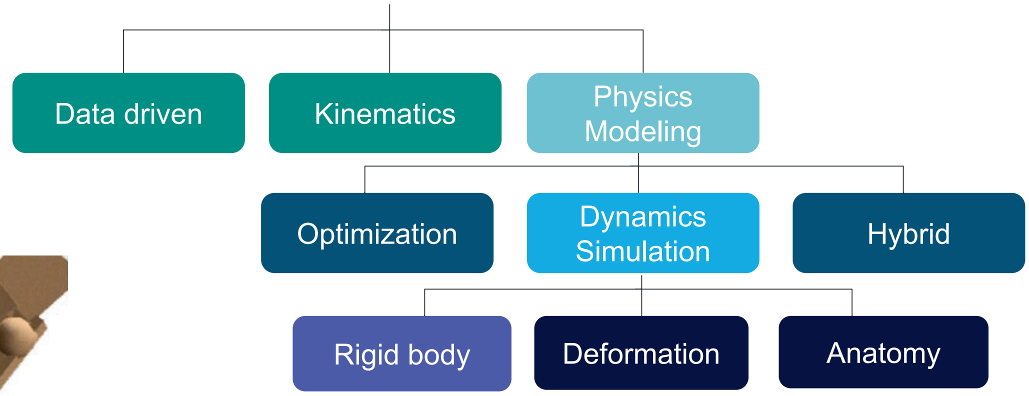
Rigid body

Deformation

Anatomy

Introduction to physical modeling for hands

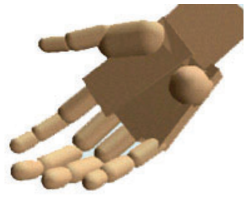
Taxonomy



Data driven
Kinematics
Physics Modeling

Optimization
Dynamics Simulation
Hybrid

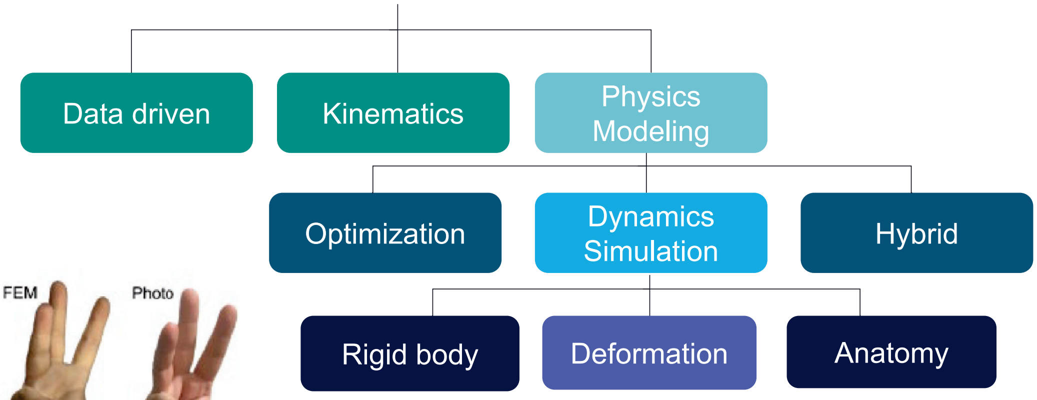
Rigid body
Deformation
Anatomy



[Pollard & Zordan 2005]

Introduction to physical modeling for hands

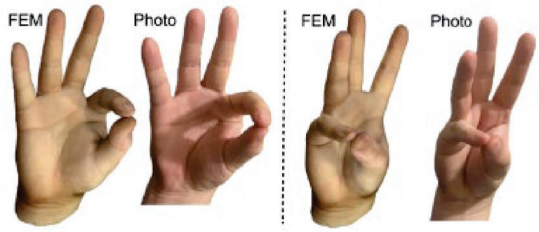
Taxonomy



Data driven
Kinematics
Physics Modeling

Optimization
Dynamics Simulation
Hybrid

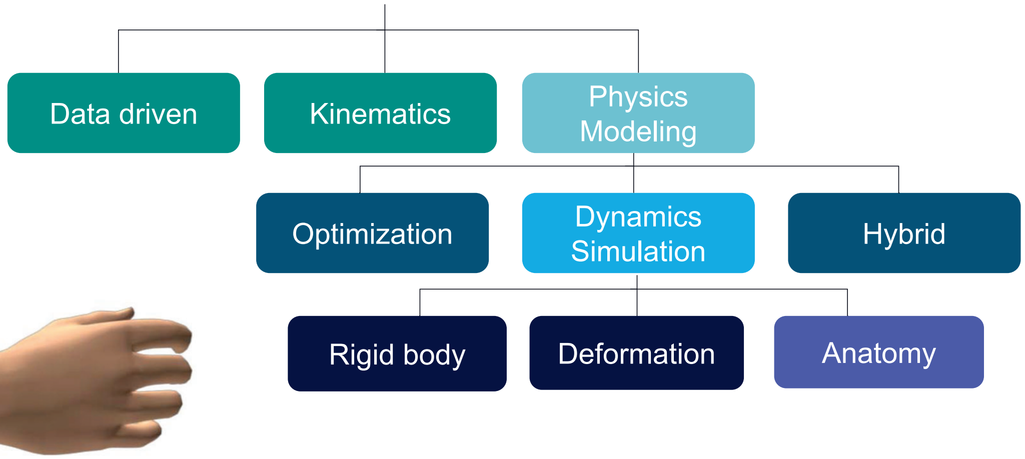
Rigid body
Deformation
Anatomy



©Wang et al. 2019

Introduction to physical modeling for hands

Taxonomy



Data driven

Kinematics

Physics Modeling

Optimization

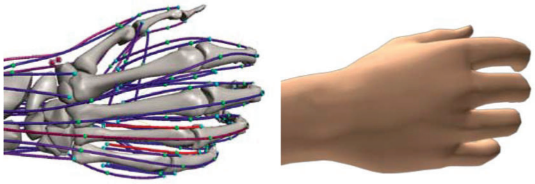
Dynamics Simulation

Hybrid

Rigid body

Deformation

Anatomy



[Sueda et al. 2008]

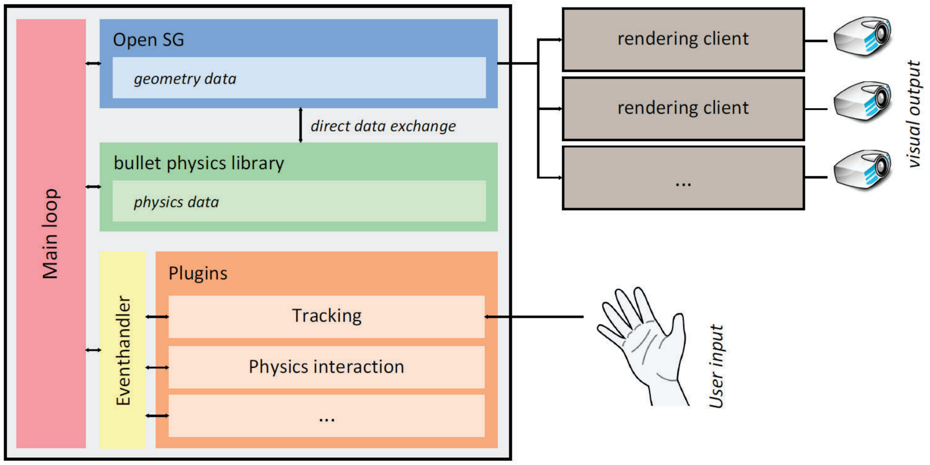
VR with physical modeling layout

Dynamics Simulation and VR

Most game engines support some physics (e.g. Bullet…)

Hand motion becomes a specia plugin input to physics library

Redo, w/ no copyright?



Main loop

Open SG

geometry data

direct data exchange

bullet physics library

physics data

Eventhandler

Plugins

Tracking

Physics interaction

...

rendering client

rendering client

...

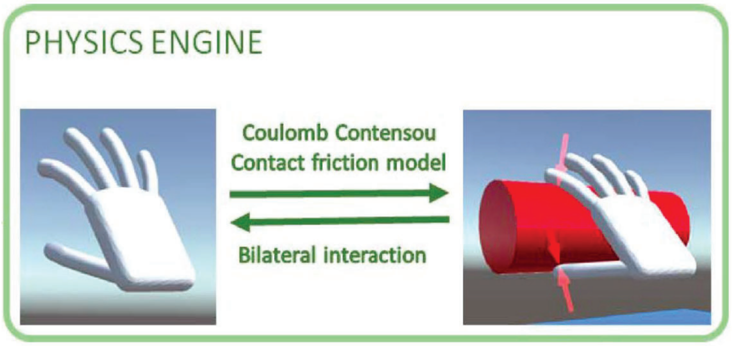
visual output

User input

©IEEE [Jacob and Froelich 2011]

Simulation – equations of motion

Rigid body formulation



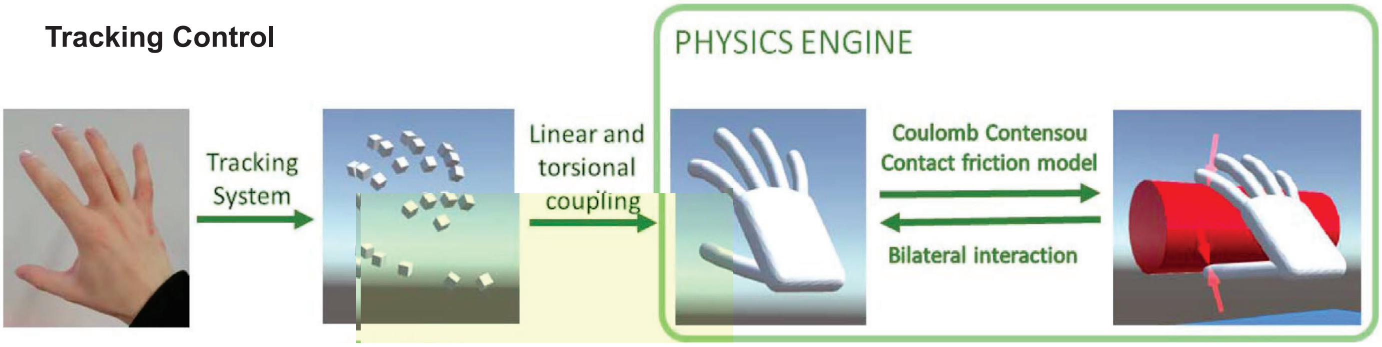
©IEEE [Delrieu et al. 2020]

$$\boxed{M(\mathbf{q})\ddot{\mathbf{q}} + h(\mathbf{q}, \dot{\mathbf{q}}) = \tau + J^T f}$$

Hand dynamics = Control + Contact

Simulation – basic controller

Tracking Control



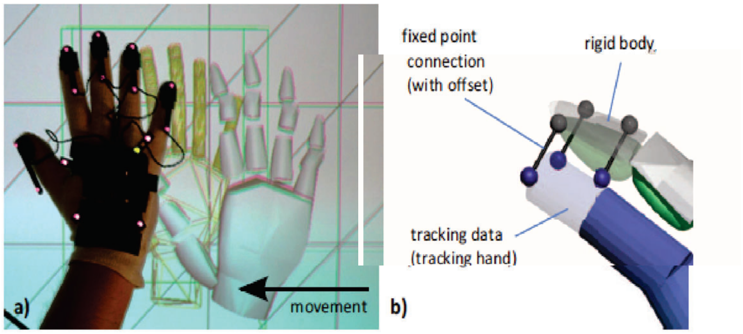
©IEEE [Delrieu et al. 2020]

$$\boxed{\tau_{joint} = k_p(\bar{\mathbf{q}} - \mathbf{q}) - k_d\dot{\mathbf{q}}}$$

Limitations of physically based hands in VR

Physical Limitations

- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations

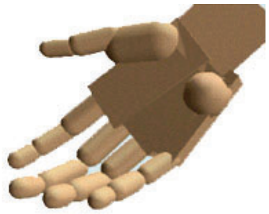

a)
movement
b)
fixed point connection (with offset)
rigid body
tracking data (tracking hand)

©IEEE [Jacobs and Froelich 2011]

Limitations of physically based hands in VR

Physical Limitations

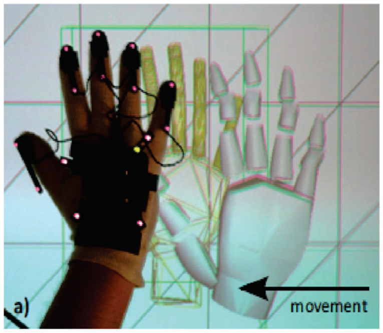
- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations



Limitations of physically based hands in VR

Physical Limitations

- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations

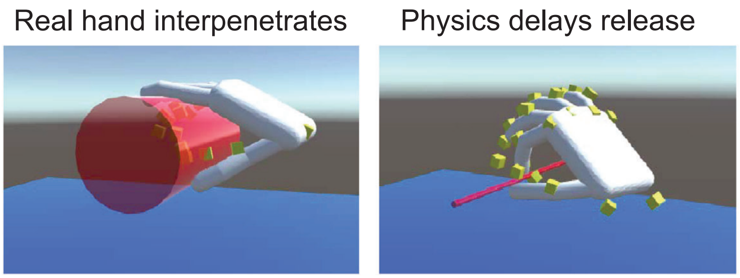


©IEEE [Jacobs and Froelich 2011]

Limitations of physically based hands in VR

Physical Limitations

- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations


Real hand interpenetrates

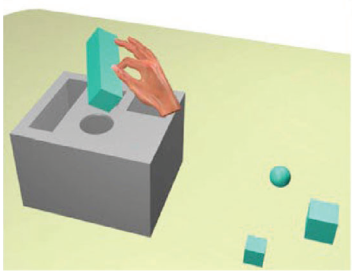
Physics delays release

©IEEE [Delrieu et al. 2020]

Limitations of physically based hands in VR

Physical Limitations

- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations



©IEEE [Tian et al. 2019]

Limitations of physically based hands in VR

Physical Limitations

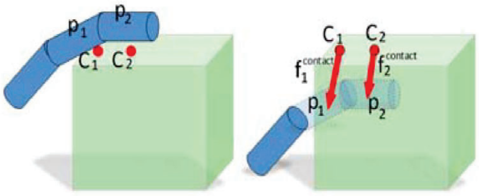
- Discrepancies between physics and user
- Tracking Latency
- Correspondence issues
- Poor grasp / contact
- Control limitations

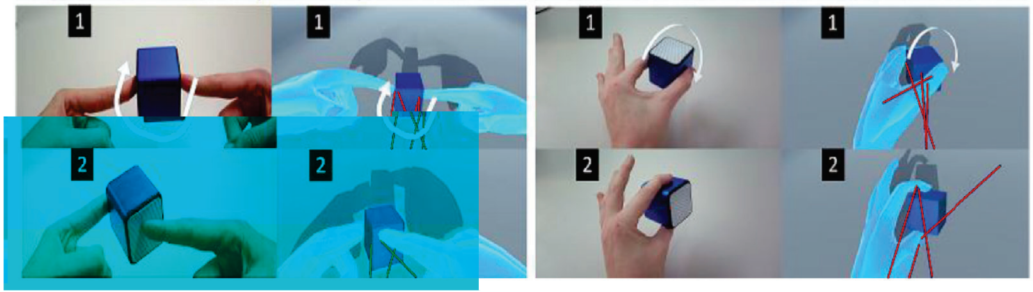
Contact modeling for physics VR hands

Contact force calculation

- Contact force, various friction models

$$\mathbf{f}_i^{\text{contact}} = \gamma(\mathbf{C}_i - \mathbf{p}_j)$$

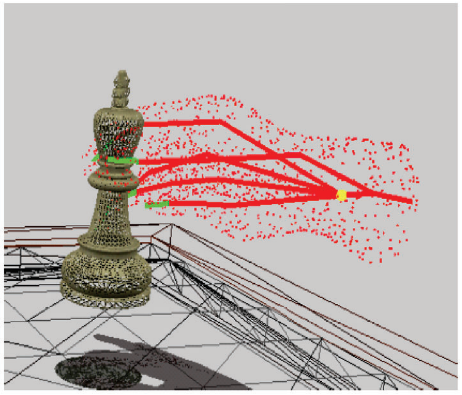


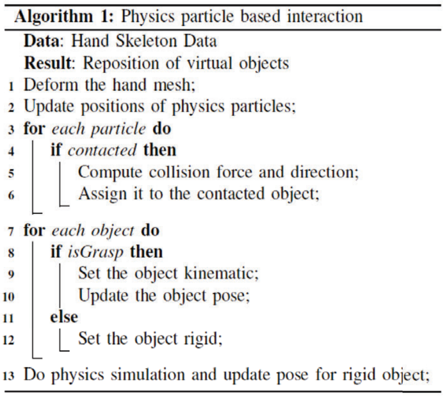


©IEEE [Holl et al. 2018]

Contact modeling for physics VR hands

Particle-based Interaction





```
Algorithm 1: Physics particle based interaction
Data: Hand Skeleton Data
Result: Reposition of virtual objects
Deform the hand mesh;
Update positions of physics particles;
for each particle do
    if contacted then
        Compute collision force and direction;
        Assign it to the contacted object;
for each object do
    if isGrasp then
        Set the object kinematic;
        Update the object pose;
    else
        Set the object rigid;
Do physics simulation and update pose for rigid object;
```

©IEEE [Kim and Park 2015]

Physical modeling with elastic tracking

Elastic tracking

- Purposeful discrepancies from real hand
- More natural interactions

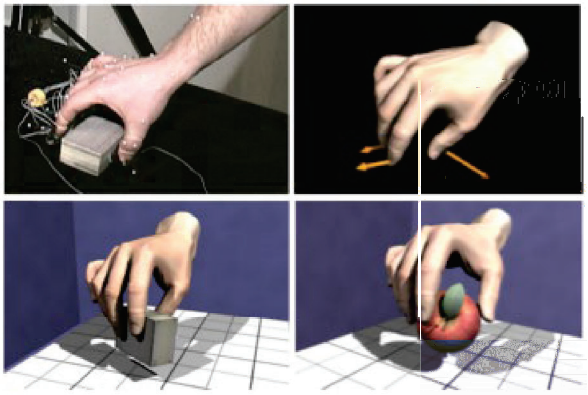
Ball grasp Pushing

©IEEE [Verschoor et al. 2018]

Physical interaction capture

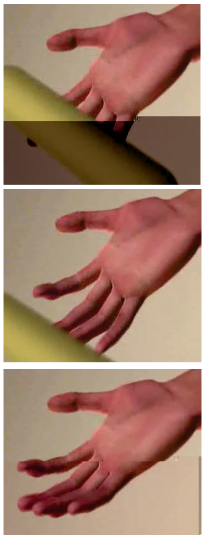
Capturing Physical Interaction

- Extending examples to new settings



[Kry and Pai 2006]

Separating passive and active control

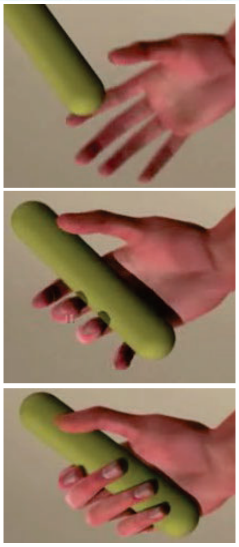


Passive Response

ligaments
tendons
skin…

Active Control

muscle activation
to achieve a task
time dependent

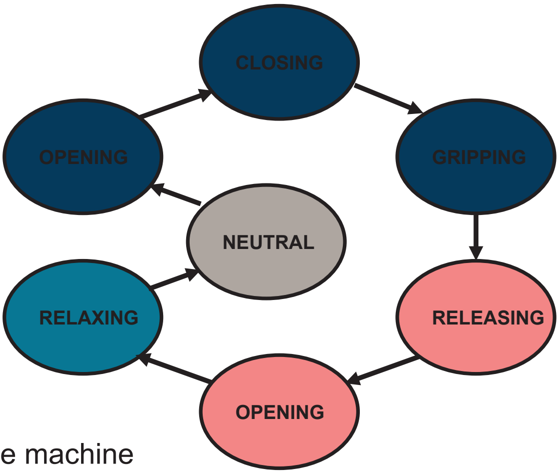


Components can easily be tuned as separate modules

[Pollard & Zordan 2005]

Controller implementation

Active Control



CLOSING
OPENING
GRIPPING
NEUTRAL
RELAXING
RELEASING
OPENING

State machine

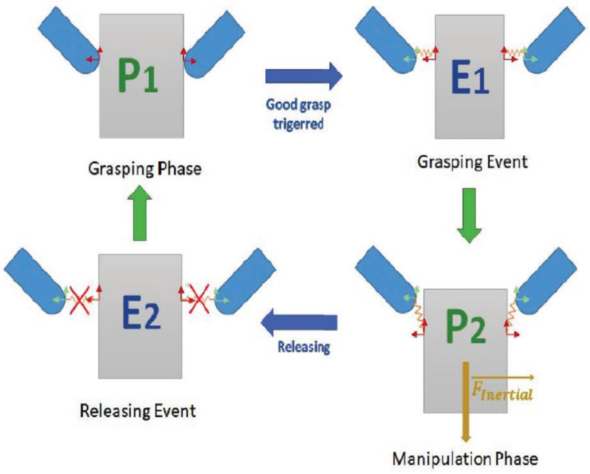
Most transitions triggered by distance from hand to object

Grasp assistant controller

State based augmentation

- Identifying phase of manipulation
- Augment grasp forces to aid behavior
- Forces account for inertial influences
- Remove forces when release is triggered

State machine for grasping

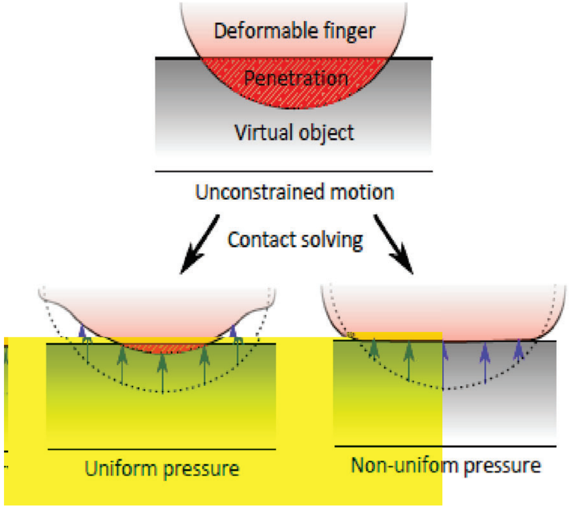


©IEEE [Tian et al. 2019]

Improving physical modeling with deformation

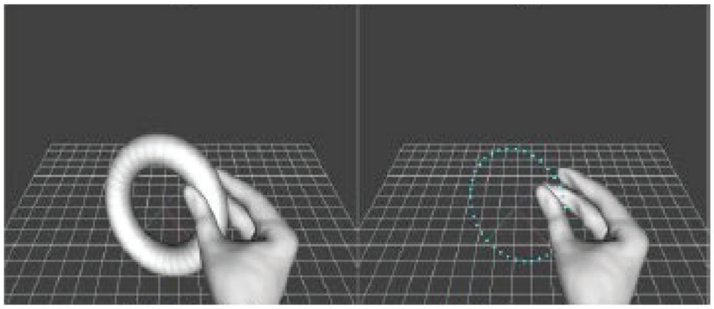
Deformation changes contact

- Deformation complies to surface
- Improved contact



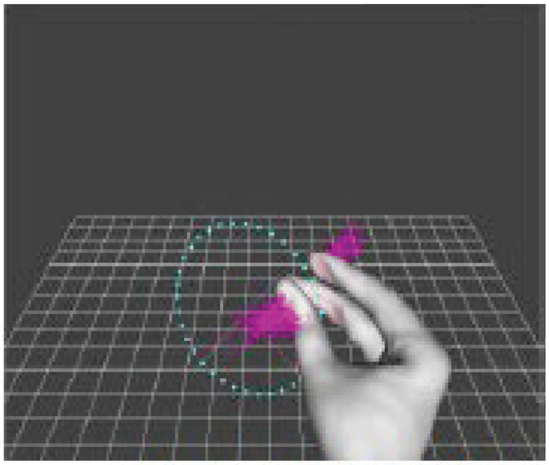
©IEEE [Talvas et al. 2015]

Improving physical modeling with deformation



Manipulation

Deformation



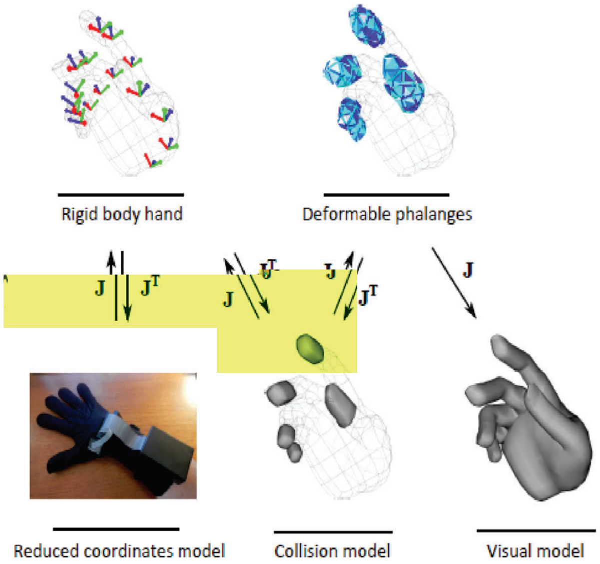
Force Contacts

©IEEE [Hirota and Tawagata 2016]

Improving physical modeling with deformation

Deformation Model to improve contact

- Deformable phalanges collide
- Increased contact surface

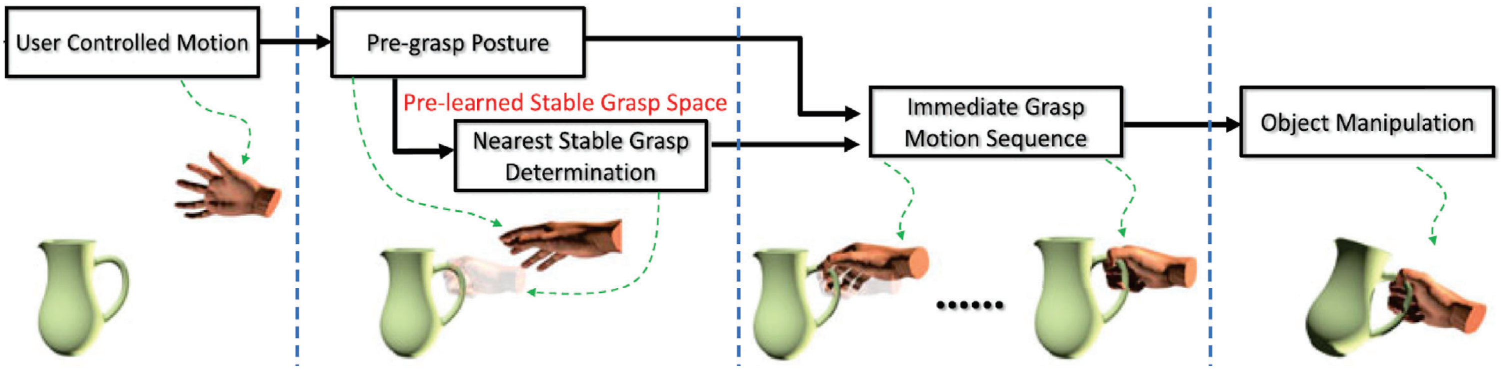


©IEEE [Talvas et al. 2015]

Hybrid solutions with physics hands in VR

Hybrid Solutions

"RTN" Realtime Hand-Object Interaction Using Learned Grasp Space for Virtual Environments



User Controlled Motion

Pre-grasp Posture

Pre-learned Stable Grasp Space

Nearest Stable Grasp Determination

Immediate Grasp Motion Sequence

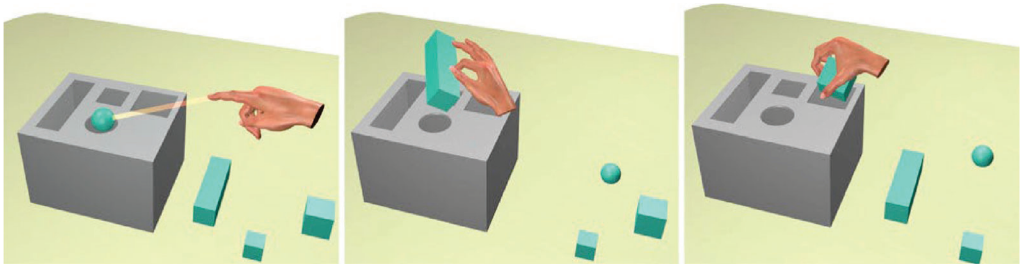
Object Manipulation

©IEEE [Tian et al. 2019]

Hybrid solutions with physics hands in VR

Challenges

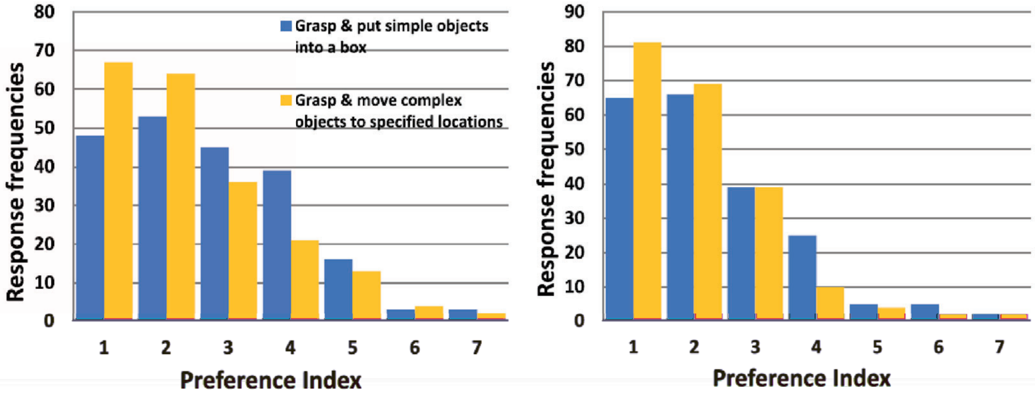
- RTN user study



Raycast grasp Pinch grasp RTN grasp

©IEEE [Tian et al. 2019]

Physical Modeling for Hands VR

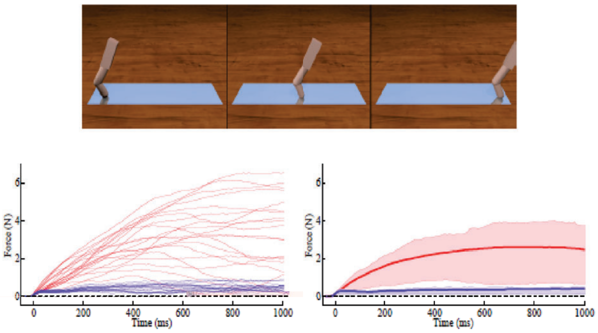

RTN grasp vs Pinch grasp

RTN grasp vs Raycast grasp

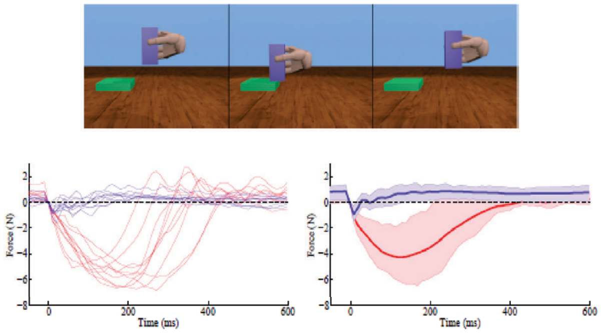
©IEEE [Tian et al. 2019]

Physical Modeling plus Haptics

Haptic feedback can augment the effectiveness of physical models



Single Participant   *All Participants*



Single Participant   *All Participants*

©Humberston and Pai [2015]

Physical Modeling - closing

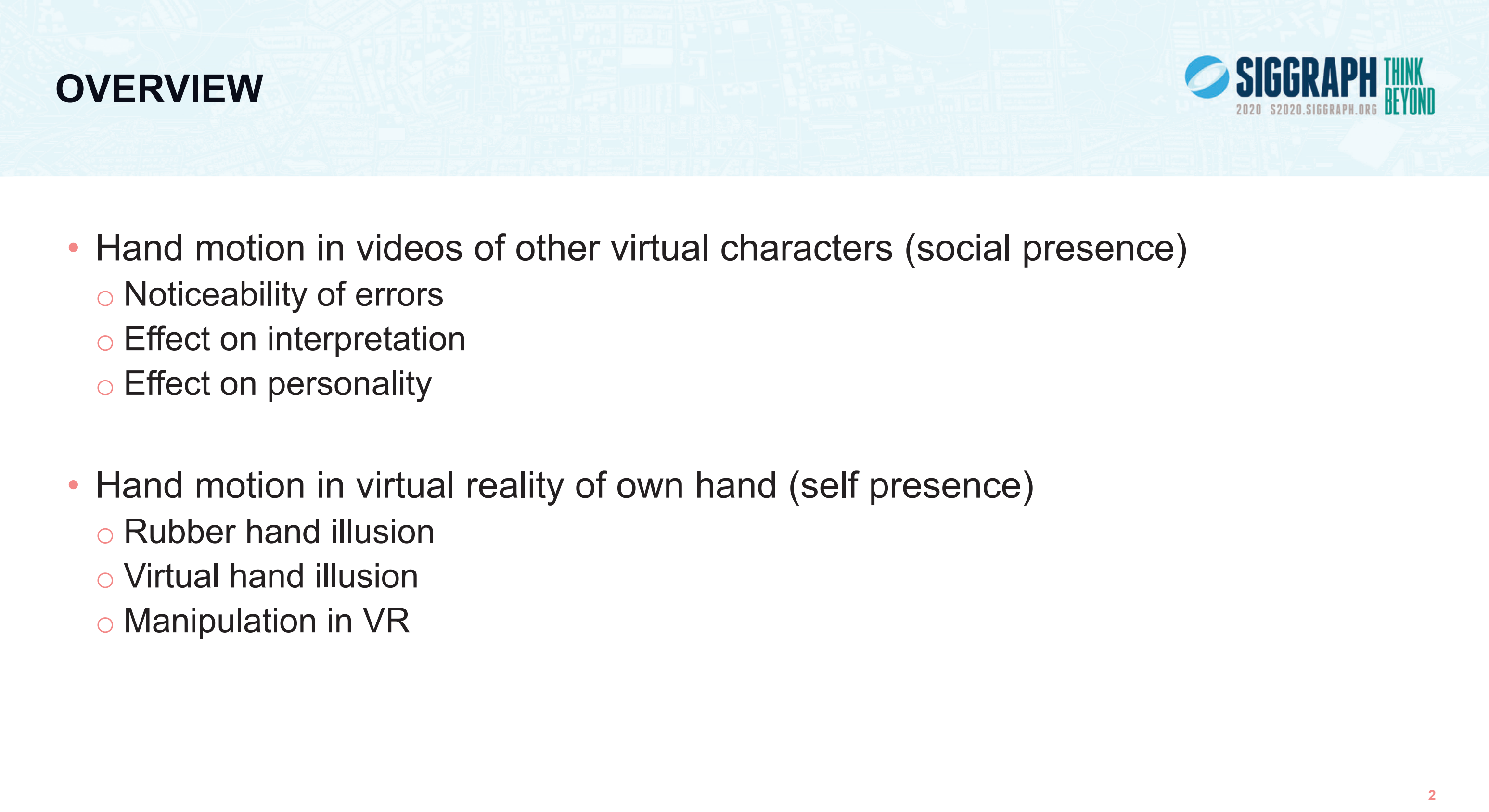
- Force-based interaction offers natural interface for VR
- Discrepancies lead to open problems between RL and VR
- Improved controllers and contact models are being explored
- Open challenges remain

SIGGRAPH THINK BEYOND

2020 S2020.SIGGRAPH.ORG

PERCEPTION

Sophie Jörg



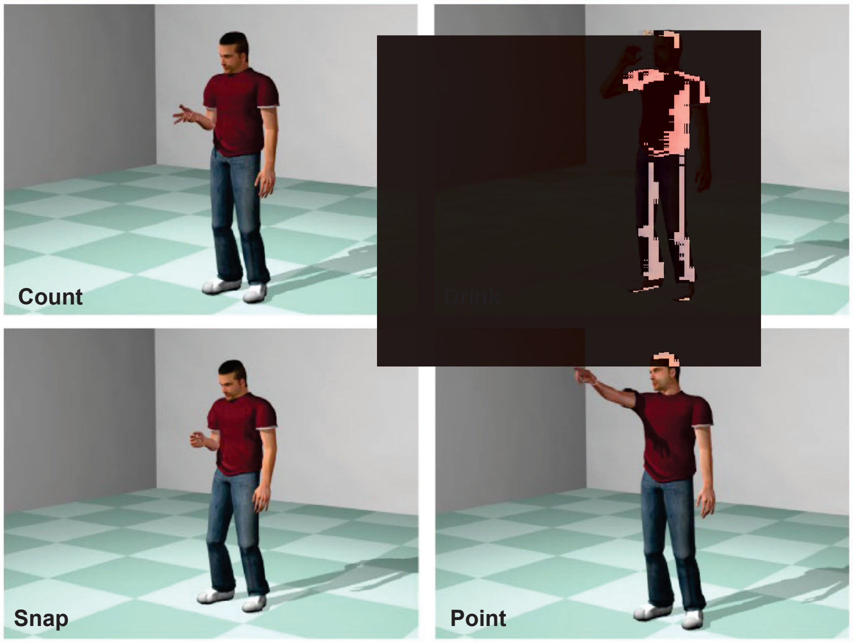
SIGGRAPH THINK BEYOND

2020 S2020.SIGGRAPH.ORG

OVERVIEW

- Hand motion in videos of other virtual characters (social presence)
  - Noticeability of errors
  - Effect on interpretation
  - Effect on personality
- Hand motion in virtual reality of own hand (self presence)
  - Rubber hand illusion
  - Virtual hand illusion
  - Manipulation in VR

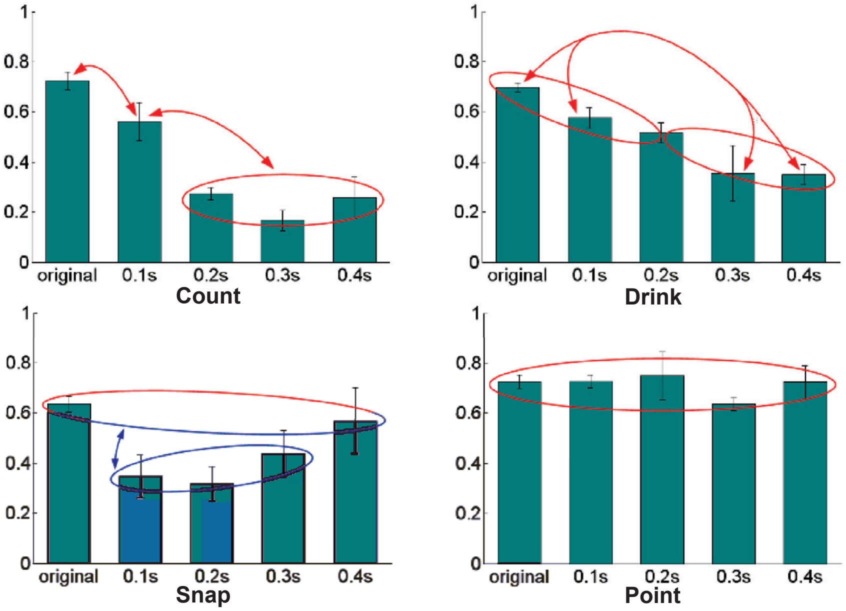
IS THIS MOTION MODIFIED OR NOT MODIFIED?



Are errors in finger motions noticeable?

***The Perception of Finger Motions*, Sophie Jörg, Jessica Hodgins, and Carol O'Sullivan, ACM Symposium on Applied Perception in Graphics and Visualization (APGV) 2010**

SUBTLE DELAYS CAN BE NOTICED



y-axes: percentage of motions rated as unmodified

Participants recognize delays as little as 0.1 seconds

However, no single threshold could be found

Can hand motion change the interpretation of a scene?

PERCEIVING INCORRECT HAND MOTIONS

Five-point scale response data: between groups ANOVA

	significant differences?
empathy questions	no
factual questions	no
quality	no

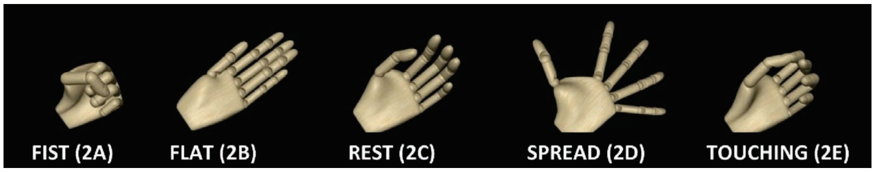
Descriptions in free form text:

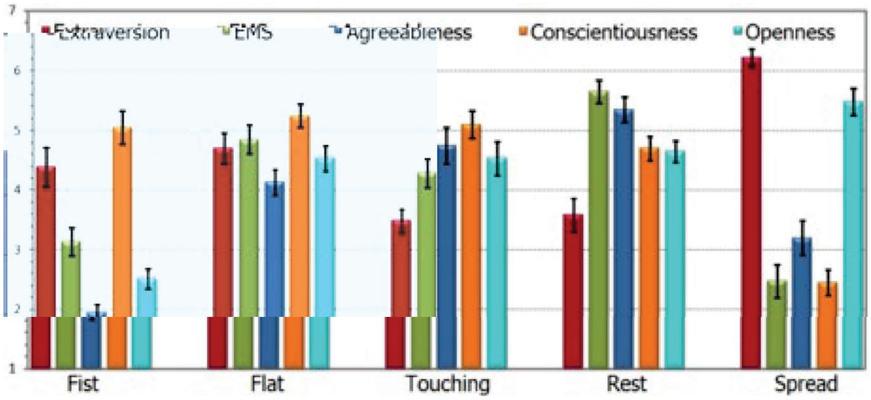
	original	*delayed*
character using ctrl-alt-del	21%	1%
computer freezing	40%	15%

INCORRECT HAND MOTIONS CHANGE INTERPRETATION

Finger motion can alter the interpretation of a scene,
even without altering its perceived quality.

HAND POSES AFFECT PERCEIVED PERSONALITY





Selected results:

- Spread most extraverted, Touching and Rest least
- Rest is most emotionally stable, Fist and Spread least
- Rest is most agreeable, Fist is least
- Spread less conscientious than all other poses
- Spread most open, Fist least

***Assessing the Impact of Hand Motion on Virtual Character Personality*, Yingying Wang, Jean E. Fox Tree, Marilyn Walker and Michael Neff, ACM Transactions on Applied Perception 2016**

RUBBER HAND ILLUSION

Participant

- Feeling that a rubber hand is part of one’s own body
- First reported by Botvinick and Cohen in 1998
- Interaction between vision, touch, and proprioception
  - Participant sits, left arm rests on a table, hidden from view
  - Rubber hand and arm place in front of participant
  - Participant looks at rubber hand
  - Rubber hand and real hand are stroked synchronously with two paintbrushes
  - A questionnaire is filled out after 10 minutes
  - Proprioception affected, proprioceptive drift

Experimenter

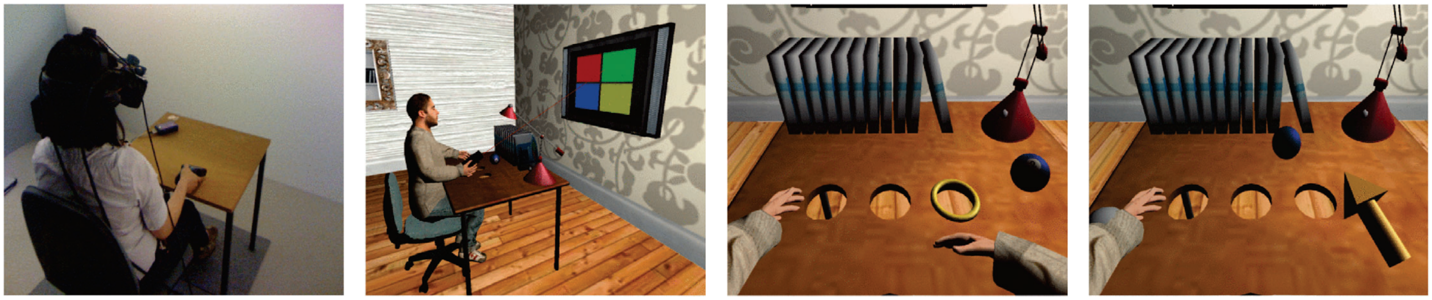
Rubber hands ‘feel’ touch that eyes see , Matthew Botvinick and Jonathan Cohen, Nature 1998

RUBBER HAND ILLUSION

VIRTUAL HAND ILLUSION

Feeling that a virtual hand is part of one's body

Induced by visuomotor feedback



© IEEE

***Is the rubber hand illusion induced by immersive virtual reality?*, Ye Yuan and Anthony Steed, IEEE Virtual Reality Conference 2010**

VHI CAN BE INDUCED FOR VARIOUS REPRESENTATIONS

© Elsevier

Participants can feel ownership for a:

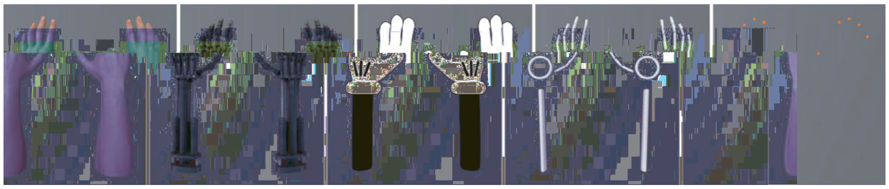
- Square or rectangle
- Balloon
- Cat claw

***Is the rubber hand illusion induced by immersive virtual reality?*, Ye Yuan and Anthony Steed, IEEE Virtual Reality Conference 2010**

VHI CAN BE INDUCED FOR VARIOUS REPRESENTATIONS

[Argelaguet et al. 2016]

© IEEE

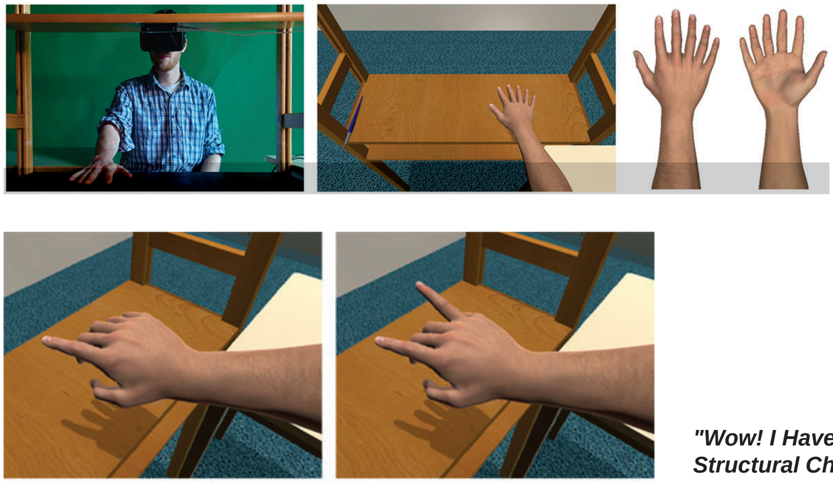


Touch with foreign hands: the effect of virtual hand appearance on visual-haptic integration, **Valentin Schwind, Lorraine Lin, Massimiliano Di Luca, Sophie Joerg, and James Hillis, ACM Symposium on Applied Perception 2018**

Participants can feel ownership for a:

- Square or rectangle
- Balloon
- Cat claw
- Abstract or iconic hands

VHI CAN BE INDUCED FOR VARIOUS REPRESENTATIONS



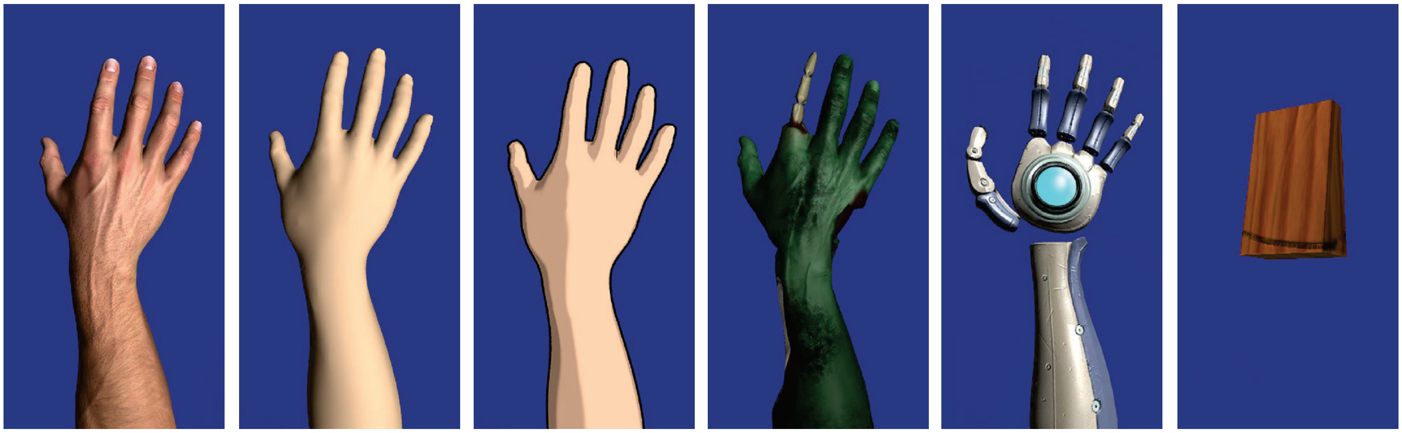
(Animated) (Rigid)

Participants can feel ownership for a:

- Square or rectangle
- Balloon
- Cat claw
- Abstract or iconic hands
- Hands with more or less fingers

***"Wow! I Have Six Fingers!": Would You Accept Structural Changes of Your Hand in VR?*, Ludovic Hoyet, Ferran Argelaguet, Corentin Nicole, and Anatole Lécuyer, Frontiers in Robotics and AI 2016**

VHI CAN BE INDUCED FOR VARIOUS REPRESENTATIONS



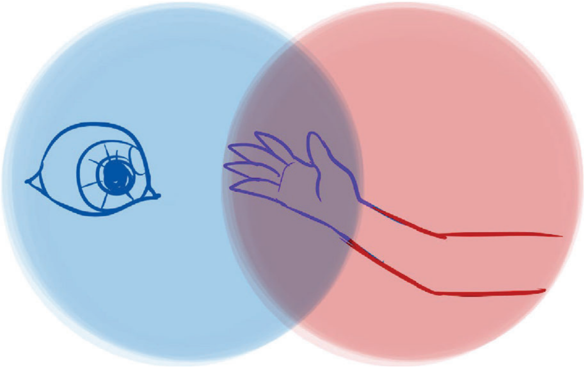
[Lin and Jörg 2016]

Strongest effect → Weakest effect

In direct comparison, anthropomorphic models lead to a stronger illusion and a realistic human model leads to the strongest effects.

TOP-DOWN AND BOTTOM-UP PROCESSING

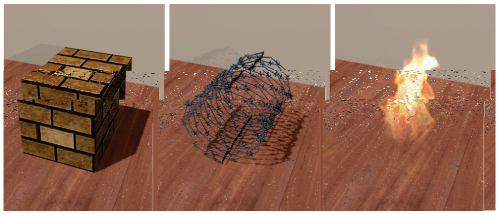
Top-Down Processing
(contextual information)



Bottom-Up Processing
(sensory input)

HOW DO WE MEASURE BODY-OWNERSHIP ILLUSIONS?

- Self-reports, questionnaires (typically using Likert scales)
  - Body Ownership, e.g., “Sometimes I felt as if the virtual hand on the screen was my own hand”
  - Agency, e.g., “The movements of the virtual hand on the screen were caused by myself”
  - Self-location, e.g., “It sometimes seemed my own hand was located on the screen”



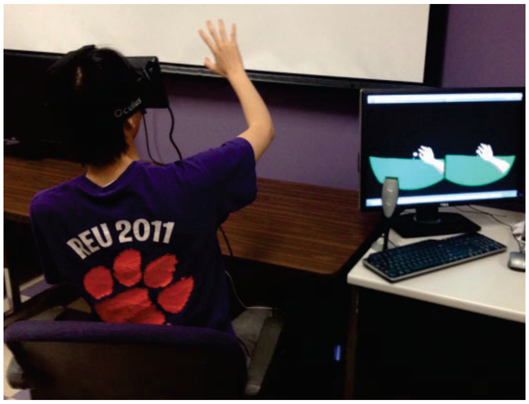
- Perceived position of the own hand, proprioceptive drift
- Reaction to threat
  - Avoidance, time to complete task
  - Skin conductance response

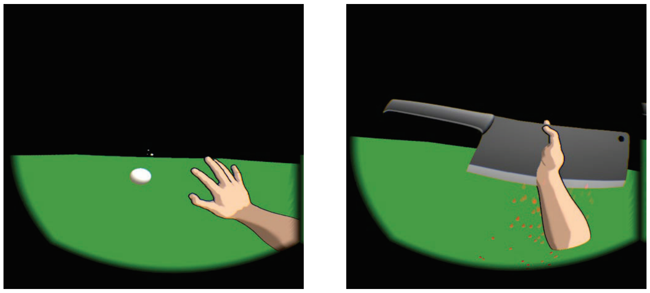
***The role of interaction in virtual embodiment: Effects of the virtual hand representation*, Ferran Argelaguet , Ludovic Hoyet, Michael Trico, Anatole Lecuyer, IEEE Virtual Reality (VR) 2016**

EXAMPLE EXPERIMENT DESIGN AND PROCEDURE

- Within-groups design, 15 participants
- Real rubber hand illusion pre-test

- Participants experienced each model for two minutes
- Catch task, then threat
- Participants were read statements and asked to choose a rating on the 7-point Likert scale



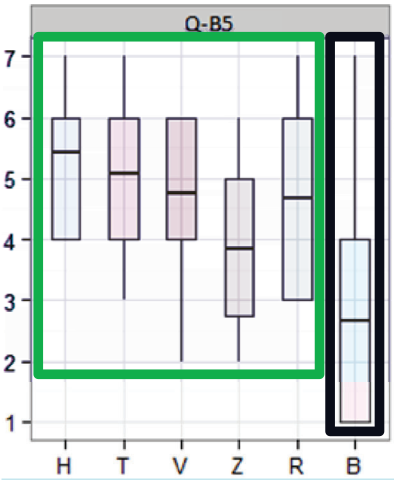


***Need a Hand? How Appearance Affects the Virtual Hand Illusion*, Lorraine Lin and Sophie Jörg, ACM Symposium on Applied Perception 2016**

EXPERIMENT RESULTS

- **Q-B5**. Sometimes I felt as if the virtual hand on the screen was my own hand. **(Ownership)**

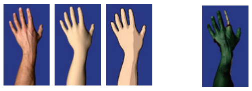
- **H, T, V, Z, R > B**



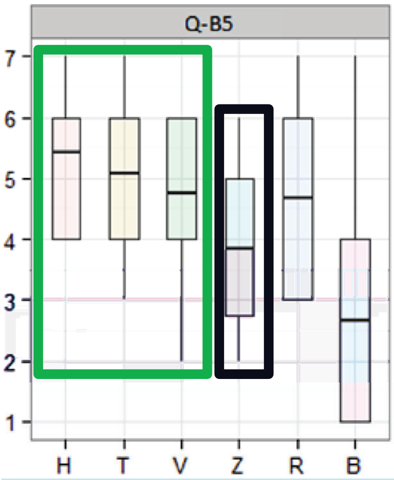
The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

EXPERIMENT RESULTS

- **Q-B5**. Sometimes I felt as if the virtual hand on the screen was my own hand. **(Ownership)**



- **H, T, V > Z**

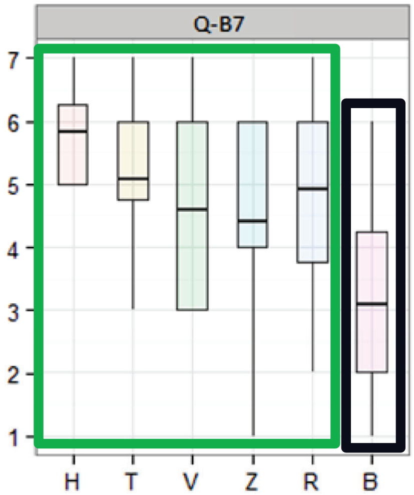


The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

EXPERIMENT RESULTS

- **Q-B7**. During the experiment there were moments in which it seemed that my own hand was catching the ball. **(Ownership)**

- **H, T, V, Z, R > B**

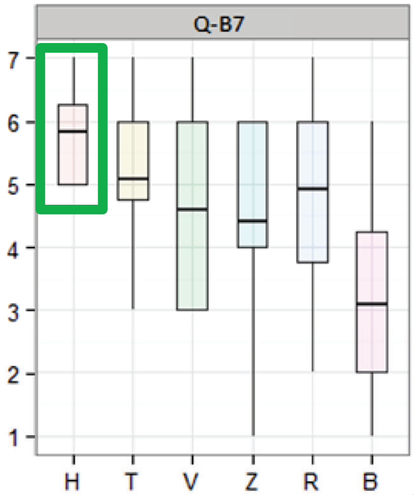


The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

EXPERIMENT RESULTS

- **Q-B7**. During the experiment there were moments in which it seemed that my own hand was catching the ball. **(Ownership)**

- **H > V, Z, B**

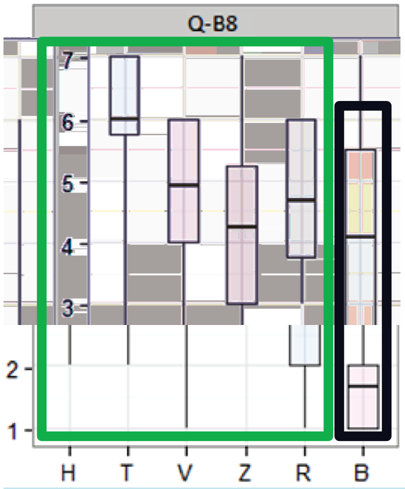


The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

EXPERIMENT RESULTS

- **Q-B8**. I thought the virtual hand on the screen looked realistic. **(Realism)**

- **H, T, V, Z, R > B**

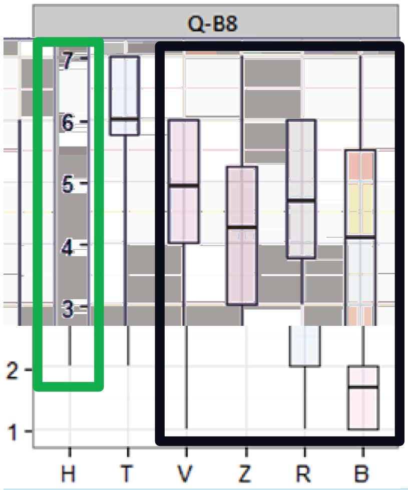


The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

EXPERIMENT RESULTS

- **Q-B8**. I thought the virtual hand on the screen looked realistic. **(Realism)**

- **H > V, Z, R, B**



The boxes indicate inter-quartile ranges and the bars show the range of the ratings.

GRASPING FEEDBACK

Virtual Grasping Feedback and Virtual Hand Ownership, Ryan Canales, Aline Normoyle, Yu Sun, Yuting Ye, Massimiliano Di Luca, and Sophie Jörg, ACM Symposium on Applied Perception 2019